DRAGON USER

The independent Dragon magazine

September 1988

Contents

Editorial

THE leading news this month is that another Dragon show will be held this autumn, in Weston-super-Mare. The Colour Computer Convention will be organised by Dragonfire Services held on a Sunday so that users who normally work on a Saturday will have a chance to attend. See Newsdesk for further details. However, Dragon User has now heard from two inside sources that the rumour of an all-Dragon show to be organised by New Era Publications was founded on hope rather than agreement and will not come to pass.

After the interest shown in the Dragon's past in recent letters pages, I would like to hear from anyone who has historical material about the Dragon, or a good collection of old DUs or just a good memory.

Meanwhile, thanks to the Arcade Arena volunteers. The column has dropped out this month for reasons of space, but should be back in the next issue.

Telephone number
(01) 570 8335

Editor
HELEN ARMSTRONG

Production Editor
HELEN ARMSTRONG/ARTSET

Administration/Advertising
BOB HARRIS

Publisher
DRAGON PUBLICATIONS

Subscriptions
UK £14 for 12 issues
Overseas (surface) £20 for 12 issues

ISSN 0265-177

Address: Dragon Publications, 49 Alexandra Road, Hounslow, Middlesex TW3 4HP, United Kingdom.

Published by Dragon Publications 1988

Typeset by Artset Limited, London NW1

Printed by Headley Brothers Ltd. Ashford, Kent

Registered at the Post Office as a newspaper. Dragon and its logo are trademarks of Eurohard Ltd.

How to submit articles

The quality of the material we can publish in *Dragon User* each month will, to a very great extent depend on the quality of the discoveries that you can make with your Dragon. The Dragon computer was launched on to the market with a powerful version of Basic, but with very poor documentation.

Articles which are submitted to *Dragon User* for publication should not be more than 3000 words long. All submissions should be typed. Please leave wide margins and a double space between each line. Programs should, whenever possible, be computer printed on plain white paper and be accompanied by a tape of the program.

We cannot guarantee to return every submitted article or program, so please keep a copy. If you want to have your program returned you must include a stamped addressed envelope.

Letters

This is your chance to air your views — send your tips, compliments and complaints to Letters Page, *Dragon User*, 49 Alexandra Road, Hounslow, Middx TW3 4HP

Pamcalls

THANK you for advising me of the smashing news that *Dagon User* is to continue under new management. My own news is similar but different.

Needing to make a contribution to the family budget, Pamcomms Ltd. for was formed well before the Dragon came on thes cene. Although I have been involved with the Dragon (my first and still best-loved home computer) since 1982, primiarly because I go not have the imagination for creating computer games the Dragon has remained a private computer interest, though also the one that has given me the greatest enjoyment. Earnings over the last three years ahave been sustained by contracts to convert games to a French 6809 for a software hosue. Alas, that market seems to be closing down, so I must seek new pastures. Although *Formula One* has been my best seller (and I am talking of 120 copies, not multiple hundreds!), it is obvious that the Dragon will not be a viable financial proposition for me. A software publisher, having seen *Formula One* at the London show, *gave* me a 16-bit machine to develop software on. To date, I have been allowing myself to be drawn back to my much loved and familiar Dragon rather than concentrating on the new beast. However, your letter suggesting that the way was now open to continue *Pamcodes* forces me to face reality and say sorry, but I must put my future energies fully into 16-bit technology. I am sad that I couldn't complete the machine code series more thoroughly, and that I haven't developed for the Dragon all the software that I would oike to, and that I haven't been to an Ossett Show.

As for the future of the Dragon, it remains a very good computer; there are new software developers replacing the drop-outs; *Dragon User* will contintue and flourish if owners realise that if they put their hands in their pockets every now and then, Dragon life is sustainable ... and I don't know that I will be able to totally divorce myself from it ...

Meanwhile, I will sign off by thanking my customers for their support. I feel even greater thanks are due to all *Dragon User* readers who have troubled to write to me about everything and anythings over the years, finally special thanks to *Dragon User* itself. Had computer programming not been such a lucrative career, I would have switched to journalism in my youth — thank you for allowing me to have my cake and eat it!. Long live the Dragon!

Pam D'Arcy
21 Wycombe Lane
Wooburn Green, High Wycombe, Bucks HP10 0HD

AND thank you for everything, Pam. I hope business continues to flourish.

But, said I to to an engineer of my acquaintance, recalling what Paul Grade said in last month's *User*, is it true that people will buy a new computer just because the colour is better and the memory is bigger, whether they need it or no? (I am not quite as naive as I sound, but, not being of this school of thought myself, I wanted a second opinion.) Oh yes, said he. There are good reasons as well, of course, but the more you find out, the more you find that people don't buy quality, they buy appearances.

Here we have an expert and dedicated software writer being dragged kicking out of the Dragon arena by financial pressures as a direct result of all those people who bought 16-bit computers because they couldn't work out what to do with their 8-bit ones. Being as I am in a position to observe a large body of unusually devoted 16-bit users fairly closely, I can report that, despite the best intentions, many of them still don't know which way up to hold the thing. Heaven knows how the off-the-peg-ST crowd copes.

Every month we will be shelling out a game or two, courtesy of our supplies, to the reader/s who send the most interesting or entertaining letters. So send us your hints and your opinions, send us your hi-scores and suggestions. Send us your best Dragon stories. What d'you think we are, mind readers?!

EXTRA PUFF

Easy is best

"AND some have greatness thrust upon them."

I was surprised to find my name in print in Gordon Lee's column in July. I am not the right sex, shape or age for Page 3, and as a convinced republican would refuse to have my name in the Court Circular, so that only leaves *Dragon User* as a remaining target!

However, I am not writing to preen my expanded ego, but to confirm Gordon's wise advice about a useful textbook. By a strange coincidence, I purchased the *Easy Programming* book he recommends at about the same time as I wrote to him, and can certainly confirm that it is a most helpful and well-written book. I now agree with his emphasis on knowing thoroughly the action of each command. I was tempted to skip the first 'baby' chapters, but, having decided to work through them, found that even the simplest excercise had been carefully thought out to teach a lesson and that much of my previous difficulties were because I had only partly understood certain functions.

However, Gordon Lee is in no need of endorsment from me, and my main reason for writing is to let readers know that the book is available from R & AJ Preston, Kings Hall Court, St. Brides Major, Mid. Glam CF32 0SE. I think that the second book is, too, but do not have their list at present to confirm. However, they will no doubt be glad to let anyone interested have a copy of their attractive little catalogue of games and books.

Jim Finlay, Romford

I haven't got all the details at time of writing, but Preston's are glad to receive enquiries. There should be an advertisement in this issue with details of their new games. The very same advertisement missed last month's issue because of a postal dispute in some North London sorting offices, so they deserve an extra plug.

I'm short of ready-to-publish letters this month owing to the abovementioned moves, holidays and other chaos. I have a large packet still to sort, so hopefully we will be back to normal next month.

Maths on the run

AN answer (perhaps only partial) to Pal Dahle's query (*Input on the run?* July 1988) reference putting in mathematical functions to a running program may lie in the key part to a program for drawing graphs which I submitted to DU some years ago but alas! It was rejected. The relevant part of the larger program is appended.

The function is entered as a string (F$) which is analysed for trig functions, operators etc., which are then tokenised and poked (preceded by the tokens for DEF FNX(X)) into a 'reserved' program line (400), the whole line, or its remainder after poking in the function, being made inoperable by inclusion of the token for REM. The position of the reserved line is variable ST. A subroutine (360) to restore the whole reserved line to a REM statement is included, otherwise every time a new function is entered, the program would have to be reloaded. Note that the 'reserved' line must be at least 10 characters longer than any function which is to be entered to allow for DEF etc.

F.G. Holliman
6 Kings Grove
Lingniddry
East Lothian
Scotland EH32 0QW

PS The '730' mentioned in line 430 is the line in the main program when the entered function is used.

Inputting a maths function to a running program:

```
90 GOTO390
100 REM*****SUBROUTINE TO TOKENISE FUNCTION AND POKE TO
 PROGRAM LINE*****
110 B$="SQRSINCOSTANLOGEXPART"
120 FOR I=1 TO 7:READ B(I):NEXT
130 C$="+-*/^":FOR I=1 TO 5:READ C(I):NEXT
140 DATA 133,136,137,138,134,135,139
150 DATA 195,196,197,198,199
160 DATA 152,32,190,88,40,88,41,203
170 FOR I=0 TO 7:READ X:POKEST+I,X:NEXT:RESTORE
180 J=0:FOR I=1 TO LEN(F$)
190 IF LEN(F$)>2 THEN T$=LEFT$(F$,3) ELSE GOTO220
200 B=INSTR(1,B$,T$):BA=(B+2)/3
210 IF B<>0 THEN GOSUB270:GOTO250
220 T$=LEFT$(F$,1)
230 B=INSTR(1,C$,T$)
240 IF B<>0 THEN GOSUB320:GOTO250:ELSE POKEST+7+I-J,ASC
(T$):F$=RIGHT$(F$,LEN(F$)-1)
250 NEXT I:GOTO350
260 REM***POKE TRIG FUNCTIONS SUBSUBROUTINE***
270 POKEST+7+I-J,255:POKEST+8+I-J,B(BA)
280 F$=RIGHT$(F$,LEN(F$)-3)
290 I=I+2:J=J+1
300 RETURN
310 REM***POKE OPERATOR TOKENS SUBSUBROUTINE***
320 POKEST+7+I-J,C(B)
330 F$=RIGHT$(F$,LEN(F$)-1)
340 RETURN
350 POKEST+7+I-J,58:POKEST+8+I-J,130:RETURN
360 REM***POKE TO CLEAR FUNCTION LINE SUBROUTINE***
370 POKEST,130:FOR I=1 TO 73:POKEST+I,42:NEXT:RETURN
380 REM*****MAIN PROGRAM*****
390 ST=256*PEEK(166)+PEEK(167)+20:GOSUB370:GOTO420
400 REM*****************************************************
**************************
410 RETURN
420 CLS:PRINT"ENTER THE X-PART OF YOUR         FUNCTION
(NO 'Y=' REQUIRED)     USING STANDARD DRAGON SYNTAX
(E.G. 'TIME' IS '*' NOT 'X' AND POWERS ARE PRECEDED BY
↑";
430 PRINT:PRINT"if 'error in 730' appears, the functio
n has been entered         incorrectly.  rerun the progra
m and amend the entry"
440 PRINT:INPUT"YOUR FUNCTION";F$:E$=F$
450 IF LEN(F$)>64 THEN PRINT "FUNCTION TOO LONG": FOR D
=1 TO 1000:NEXT:GOTO440
460 GOSUB 110
```

Here are the solutions to Crosswords three, four, five and six:

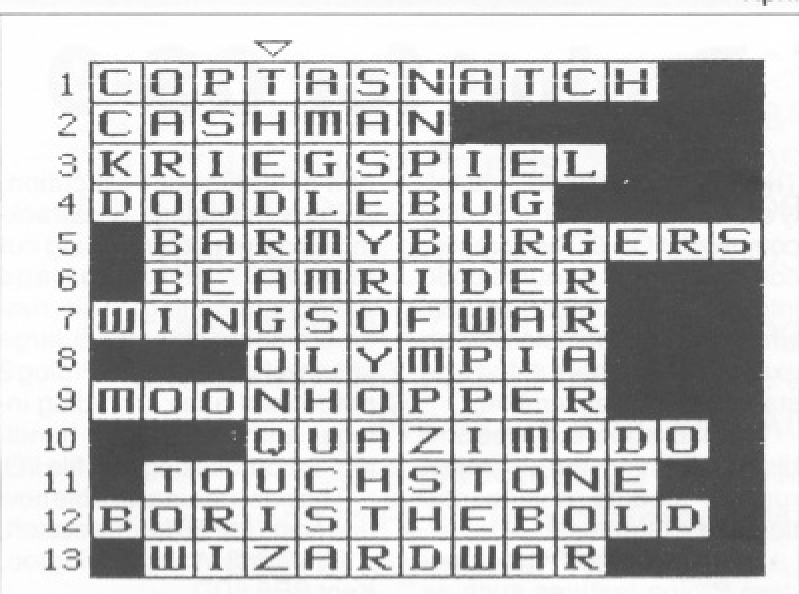
February

1 DEMONSEED
2 GRIDRUNNER
3 FRANKLINSTOMB
4 GRABBER
5 VORTEXFACTOR
6 MINEDOUT
7 DEFENCE
8 NORTHSEAOIL
9 REDMEANIES
10 SPEEDRACER
11 CAVEHUNTER
12 TREKBOER
13 DRAGONCHESS

March

1 SYZYGY
2 MODULEMAN
3 FROGGER
4 ATHLETYX
5 SWASHBUCKLER
6 ALCATRAZ
7 RUBYROBBA
8 THEKETTRILOGY
9 BANDITO
10 PHANTOMSLAYER
11 TIMEBANDIT
12 ESCAPE
13 COSMICCRUISER

April



May

1 BUBBLEBUSTER
2 MANICMINER
3 BERSERK
4 CHICKENRUN
5 HUNCHBACK
6 TELEWRITER
7 TOTALECLIPSE
8 DUNGEONRAID
9 DRAGONHAWK
10 BOULDERCRASH
11 TANGLEWOOD
12 NIGHTFLIGHT
13 ENO

If you have any new products for the Dragon — software or hardware — please write to The Newsdesk at 49 Alexandra Road, Hounslow, Middx TW3 4HP

6809 Show goes to Weston-S-M

Dragonfire Services are to organise a show for the Dragon and Tandy Colour computers at Weston-super-Mare, Avon, on December 4th 1988.

John Penn of John Penn Discount Software is quoted as saying that it is unlikely that the Penns could organise a London show this autumn, owing to the very high cost of London venues. Weston-super-Mare, near Bristol, has excellent road and rail access to most of the UK and is fairly local to the organisers.

'All the major Dragon and Tandy suppliers will be there with software, hardware and supplies. There will be special show reductions, and new software will be released at the show. There will be bargains.' say Dragonfire. Computape, John Penn Discount Software, Orange Software, Pulser Software, R & AJ Preston, NDUG and Dragon Magazine have already put down their names to attend.

The Colour Computer Convention will be held at the Arosfa Hotel, Lower Church St., Weston-super-Mare from 10am to 3pm on Sunday 4th December, entrance £1.50 (OAPs/under 10s 75p; under5s/disabled free).

Dragonfire also write "Please print our address in the supplier column in reviews, as we had orders as a result of the *Computa-text* and *Script* review, but none from the *Pyradventure* and *Underbeings* review." In future, all reviews will contain the supplier's address, but for the above games and information about the show, contact Dragonfire Services, 13 Parry Jones Close, Blaina, Gwent NP3 3NA.

Anyone who finds that *Comptuta-text* does not function fully with their version of *Printer Control* should contact Dragonfire.

New Era moves into software

NEW Era Publications, publishers of *6809 User* (formerly *Dragon's Roar*) are launching a new software label, New Era Software, in order to endorse their commitment to all areas of Dragon publishing.

New Era is presently looking for software writers, offering a royalty rate of over 35%. Contact New Era at 37 Collins Meadow, Harlow, Essex CM19 4EN.

Ink and Ink again

From John Smallwood

A firm called Aladdink will re-ink used printer and typewriter ribbons. The first re-inking costs £1, and the company will enclose an estimate with the returned ribbon for how much it will cost to re-ink the ribbon in the future.

This is a helpful alternative to seeking out obscure or obsolete ribbons (see *Letters*, July 1988). Write to Alladink (Dept. 80), 4, Hurket Crescent, Eyemouth, Berwickshire TD14 5AP. Tel. 08907 50965.

Dragon User would welcome a consumer report from anyone who has used this service.

Extension keyboards from German source?

From David Rothery

A source has been sighted in Germany which apparently supplies a replacement keyboard and interface, featuring 94 keys including 10 function keys, and a integral real-time clock on the circuit board.

The add-on board must be soldered inside the Dragon. The clock carries the date and time permanently and can be accessed under DragonDOS for inclusion in accounting programs, etc.

The clock's most useful function is with OS-9, where it datestamps each file it saves and will automatically date letters written using Stylo and Mailmerge.

A new CLOCK unit is supplied for the bootfile, so that when a GETIME system call is made, the new clock is used. The keyboard drivers are supplied, along with some DragonDOS software.

The package costs £50 plus £2.50 p&p' the real time clock by itself costs £33 plus £2.00 p&p.

Dragon User has contacted the address supplied, Seigfrieds Schraubenzieher, c/o Alexander Goeschel, Grafstrasse 2 D-8523, Baiersdorf 1, Federal Republic of Germany, for confirmation and literature, but has had no reply at time of going to press. There is said to be a 'limited number of keyboards' and that 'fitting can be arranged', but no further details are given.

Maplin Electronics still in the Dragon business

THREE hardware constructional projects for the Dragon are available from Maplin Electronics: the Dragon 32 Extendiport (pcb only, £3.80), the Dragon 32 RS232 Modem Interface (pcb only, £3.95, kit £13.95) and the Dragon 32 i/o Port (pcb £5.50, kit £17.95). Constructional details can be found in Maplin Project Book 10 (the Extendiport) and Maplin Project Book 8 (the other two). The project books are 85p each.

These details are taken from the current Maplin catalogue, page 293. Maplin can be contacted at PO Box 3, Rayleigh, Essex SS6 8LR, Tel. Southend-on-Sea (0702) 552911 (mail order) and has shop (non-mail-order) in Manchester, Birmingham, Bristol, Westcliff-on-Sea, Southampton and London.

Prolog for OS-9

The new product from Chris Jolly's firm Metasoft is a Prolog compiler for OS-9. The compiler converts Prolog 2 source code into 6809 assembler source, which can be assembled into an executable program using trhe standard OS-9 assembler.

The package is supplied on a disc contianing the compiler, runtime module, documentation and examples.

The compiler has all the standard Prolog features such as non-deterministic execution, pattern matching, backtracking, program control using cut and fail, recursion and metaprogramming. The runtime module includes a large subset of the standard Prolog 2 predefined rules, including integer arithmetic, string handling, list processing and file I/O.

The package is available now and costs £12.50 from Metasoft, 4 Pinehurst Walk, Orpington, Kent BR6 8DD.

New software for review should be sent to *Dragon User*, 49 Alexandra Road, Hounslow, Middx TW3 4HP

The short and the long of it

Program: Visitext-Plus, Electronic Author
Supplier: *Orange Software, The Garth, Star Road, Nant-y-Derry, Abergavenny, Gwent NP7 9DP.*
Price: £13.99, £19.95

Electronic Author has been around for some time now and as far as word-processors are concerned it has had things pretty much its own way. When I found out that Orange software were bringing out a new wordprocessor called *Visitext-Plus*, I was quite keen to do a comparison. I was even more keen when I found out that Orange were about the put out *Electronic Author* V2.0. What follows is, I hope, a well balanced comparison.

Visitext-Plus

Ron Sibthorpe's original idea was to write a program that would allow him to write letters on his Dragon. However, like many a good idea it grew and grew. The end result is a WYSIWYG (nearly) word processor that uses a 64-column screen and can handle 308 lines, about three pages of text, at a time.

Visitext is certainly one of the easiest systems to learn that I have ever seen. All that is necessary is to load the disc and type BOOT and away you go. The program is predominantly menu-driven and, rather surprisingly, is a mixture of Basic and machine code. In the past Basic word-processors have been criticised for being painfully slow. This is where the machine code routines come in. Where speed is necessary, machine code is used; elsewhere, Basic rules. This makes the system easy to modify or debug where necessary. In order to conserve space, Ron has produced a modular program which keeps its routines on disc and only loads them when they are required, leaving a useful space for text even on the D32. The 64-column screen is a real gem. Those of us who use monochrome monitors or televisions will appreciate the flicker free, black on white display. By using an unusual font for this screen Ron has produced probably the most readable 64-column screen ever to be seen on the Dragon. The characters do not merge, and apart from a slight congusion with capitals S and W, it was no problem to read.

For the uninitiated I should explain that WYSIWYG is an acronym meaning What You See Is What You Get. This is what makes *Visitext* easy to use. It is possible to load the system, write a letter and print it correctly, without learning a single control code. If it is right on the screen, it will be right on paper. Where the system fails is where alomst all other systems go wrong, namely on non-standard characters like bold or enlarged. These only appear as standard characters, sandwiched between two little graphic characters which indicate control codes. Now, I will never be able to look at a vertical squiggle and know that it means bold type, or that sideways "e" means that bold has been cancelled. Still, as I said before, this failing is shared by many other so-called WYSIWYG systems. However, a more serious problem is that no matter what typeface you use, you are still stuck with 64 characters per line, nothing more, nothing less.

Regularly used lines and phrases can be stored a quick texts, which can be called using only two keys. These can be up to 64 characters ling and may contain control codes, such as new line or enlarged print. They can be held as temporary files or saved to disc. In this way it is possible to create and store an address block, which can be called up whenever it is needed. Pressing the BREAK key forces a return to the main menu, while the CLEAR key acts as a control key which when followed by any other key will enter either a printer control code or a 'quick text'. Although auto-repeat, on all keys, is used the speed can be adjusted or the feature switched off using the configure option. Unfortunately, holding down the SHIFT key forces a repeat of the character types, until the SHIFT is released. For this reason it is best to use the shift lock (SHIFT + O) to write even a short word in capitals. The program comes ready set up for an Epson FX100, or compatible, printer. However, it can be reconfigured to suit whatever machine you used. Any program claiming to be WYSIWYG would have to feature wordwrap and *Visitext* is no exception: words are never split over two lines, and unnecessary leading spaces are ignored.

Bug call

Text can be stored on, or loaded from, disc and printed in whole or in part. My review copy had a bug in the SAVE/LOAD routines which resulted in an error message and failure of the routine if the directory was accessed before saving a program. A quick call to Orange Software soon effected a cure. Similar problems occurred when using the save routine for the 'quick texts'. Although I managed to cure the problem, the curious thing is that I could not see anything wrong with the original routine. Perhaps it just did not agree with my SuperDOS.

A Move Text routine is included for either a block of text or a screen window. However, the procedure necessary to achieve this is somewhat cumbersome, and if the original is deleted the existing text is not oved up to fill the gap. The result is a hole in the text that has to be removed manually. The find and change string procedure is slightly easier to use, but subject to the same limitations. The replacement string must be the same length or shorter than the string to be replaced, if it is shorter, then gaps are left. Both these routines are of questionable usefulness, and I must admit that I feel that their inclusion is little more than window dressing. No word count or page numbering is available, and it would have been nice to have the paper wound out of the printer once the run was finished.

Documentation was quite good and well presented in its bright orange folder. However, it did tend to be a bit bague in places, especially the parts dealing with the Move Text routine and also the saving of 'quick texts'.

The ease of use offsets these idiosyncracies to a fair degree, and with a price tag of only £13.99 it seems good value. The bugs are a different matter: Graham Smith assures me that all new copies will be bug-free, and in the meantime anyone who experiences problems of a similar nature should contact Orange Software.

Electronic Author

Being an all-0machine code program *Electronic Author* occupied only 6K with another 6K allocated to the high resolution screen. There is still space for over 17K of text, even when running on a Dragon 32. It is supplied with a program called *Config*, which is used to set up the program to whatever printer you care to use. This will set up all the commonly used codes, the no-so-common ones being catered for in a different way.

My first impression of this system was somewhat mixed. Being both impatient and lazy, I expected to just RUN "AUTHOR.BIN", but it did not work like that. Even a Basic loader program currupted the program. I ended having to stick to the instructions and LOAD, then EXEC the program. In fact, this is the only different I could find between my original Smithson Computing copy and the V2.0 version supplied by Orange Software. The V2.0 version loads and runs via the BOOT command. THe main displya is on the high resultion screen and prints black on green, either 51 or 64 columns wide, with a command window at the bottom of the screen. This screen display is beginning to look pretty dated now, with many people, like myself, using monochrome monitors or black and white televisions, and it would have been better to use a black on white screen which gives a much clearer display. To achieve the 64-column screen, Wayne Smithson just removed the space between the letters. This means that you have to teach your eyes to read a new type of 'join-

ed up writing'. Because it has so many capabilities, it is necessary to read the manula supplied pretty thoroughly before starting to use the system.

Not being WYSIWYG the screen width sets no limitation on the printed width. For instance, if it is put into condensed mode the printer will print 132 columns wide and the system will handle it. What is more, if, say, a word of enlarged text is included, the program will adjust the line accordingly. Printer codes are show as initials prefixed by the control character. Unless otherwise defined, this is normally a hash sign, so # dw (*That has come out as a £ sign on your printer, Ken Ed.*)
sets double width. Such a format makes it easier to trace faults when it doesn't print correctly. A fill and justify command is available which gives nice even margins or a literal mode which allows tabulations to be achieved.

Text can be copied or deleted. Both routines either create space for the new text or close up the space if text is deleted. No separate MOVE routine is included because if text is copied to a new location and then the original deleted, then the original text has effectively been moved. A separate routine to do the job would just be a waste of space. Both page numbering and word count are supported by the program. SAVE and LOAD routines are included and, in general, work very well. There is even a facility to tag another file from disc onto the end of the text curretnly in memory, which can then be edited or moved around to form an integral part of the original article. One slilght disappointment is that the DIR command does not work with SUperDOS, though a two byte patch will cure this. Also, there is no capabil.ity to KILL a text file without leaving the program.

Conclusion

Well, the key question is, which program should you buy? The answser (as so often) depends on what you want to do with it. *Visitext* lacks the versatility and overall ability of Electronic Author. Electronic Author, on the other hand, lacks the ease of use and the outstanding screen display of *Visitext*. In writing this review, I used each system and discovered that their text files are compatible, so I will probably use both. The best advice I can give is that if you want a word processor mainly to write letters and shorter texts, then *Visitext* is for you. If you intend to go into competition with Leo Tolstay, or write your thesis, then *Electronic Author* is a better bet.

Both programs are obtainable in DragonDOS format from Orange Software for £19.95 (*Electronic Author*) and £13.99 (*Visitext Plus*).

Ken G. Smith

Electronic Author

Visitext Plus

plus *for value for money.*

Old favourite tours the world in triumph

Program: *Champions*
Supplier: Computape, 27 North End, Southminster, Essex CM0 7ND; Harry Whitehouse, 48 Queen St., Balderton, Newark, Notts NG24 3NS.
Price: £7.95

VERY few Dragon games can claim the success attributed to similar games on other computers; indeed, few Dragon games can claim to be the inspiration behind converting the idea to other computers. Yet both of these prestigious qualities form the basis of one insurmountable Dragon game, Harry Whitehouse's *Champions*.

Champions, launched in 1983, rapidly captured the imagination of many Dragon users, the result being that it secured a palace in the Dragon hall of fame as one of the most popular all-time Dragon games. And yet, while its popularity has never been in question, a review has never found its way into the pages of *Dragon User*.

To those not yet familiar, the game involves placing one in the intriguing position of a fourth division football manager whose aim is simply to become the next Brain Clough, taking the pre-selected team to the dizzy heights of the first division, and subsequently into the realms of European football.

There are indeed many realistic features incorporated within the game, many of which were updated in 1986 in a successful attempt to fight off the mounting opposition which Addictive's famous *Football Manager* impost. Such features include a transfer market, a revamped 'Bank Manager' (perhaps reminding us that football has as many battles off the pitch as on it!), a competitive and enthralling F A Cup, with limited graphics, enabling one to 'view the game in progress', while still prominently boasting the often fatal 'weekly news' feature which simulates the weeks events. Indeed, it is hard to envisage an aspect of football which the game doesn't portray.

If you ever find one of your Brand X-owning friends playing The Boss, or Soccer Boss, point out that it is a conversion of a Dragon game!

Harry Whitehouse, perhaps best still remembered in his former guise as Peaksoft, is quick to emphasise the game's success, not only in the Dragon market but more unusually in other formats as well. "The point about Champs is that it is still one of the most successful games, although few people recognise it" says Harry. "Champs did so well on the Dragon that we thought 'This can't be bad', so we re-wrote it for the ZX81, then the BBC/Electron, Tandy CoCo, Oric and Spectrum. Off it zoomed again. During this time, we'd been thinking of new features, so we did a pretty thorough re-write before we brought it out for the Commodore 64, renaming it *The Boss*. That zipped off into the top twenty, so we incorporated all the improvements into the Spectrum and BBC/Electron versions, together with a new version for the Oric Atmos, Commodore 16, MSX and Amstrad CPC."

"We still sell quite a few copies by mail order through the football magazines, but the most interesting current point is that we've licensed *The Boss* as a budget product to Alternative Software, who have released it as *Soccer Boss* for several computers. In its new clothes, it has been in the national Top Twenty for a number of months (best position so far, number two, but we're keeping our fingers crossed).

"The point about all this is that if you ever find one of your Commodore 64-owning friends playing *The Boss* or *Soccer Boss*, point out to him that he is actually playing a conversion of a Dragon game. And if he happens to have another Top Twenty game called *International Cricket* in his collection ... well, guess how that began life?"

Self evidently, *Champions* has become somewhat of a cult among Dragon users, and indeed in the computer industry in general. If the game has yet to find its way into your collection, whether you are interested in football or not, then you can be sure you will not find many more addictive and entertaining games than Harry Whitehouse's *Champions*.

Simon Jones



DRAGONSWORD!

***Paul Grade** takes a monthly stab at setting the world to rights*

JUST for a change I think I ought to try writing about a subject I know really well. Not that I don't know all about everything, of course, only our Beloved Editor could know more (and that will cost you a drink, Helen!), but I have to admit to knowing a little less about some things than others. (Such genius, and so modest too.) Anyway, the topic for today is gong to be the running of user groups and/or magazines, and there's a very good reason for this choice ... I'm getting tired of seeing groups and magazines start up, full of enthusiasm and good ideas, only to disappear without trace a few months (or sometimes weeks) later.

There's a lot of different reasons why the enthusiasm turns into disillusion, but what it usually comes down to is that people tend to get carried away with the grand concept, and don't give enough thought to the realities.

The Dragon could do with a lot more 'amateur' support. That's about the only kind it can get now, because the user base is simply too small for anyone to operate on a commercial basis, and anyone thinking they can make a profit out of the Dragon scene now probably believes in fairies, Santa Claus and election promises too.

No profit

There's a lot of scope for anyone wanting to help keep the Dragon alive, software writing, small inexpensive hardware projects, etc., and of course running magazines and groups. None of these is going to make any profit, but there's no reason for them to make a loss, either, though they all need a lot of work. If you aren't prepared to work, do everyone a favour and don't even try!

Running a group is probably the most difficult of the lot (and I'm not just saying that because I run one). The first thing you need to decide is how big you want it to be, whether it should be 'local' or 'national'. Unfortunately this is where the mistakes usually begin! There's a temptation to think TOO big and get ideas about setting up a national group or mag when resources aren't good enough. To run a group of around a thousand users you need more than just enthusiasm. For a start, you need to work out where the members are going to come from, how you're going to let them know you exist, and more to the point, what you're going to offer that will make joining your group the one offer they can't refuse.

Think you can do it by advertising? *Dragon User* and *Update* would probably be happy to give you a mention, and you could even buy advertising space, but you'd be lucky if that got you more than half a dozen replies, and 50 per cent of those would never be heard of again after the initial enquiry. Believe it or not there is NOT a crowd of Dragon owners out there holding their breath and waiting for the chance to join your group, just some very cautious and cynical ones who have lost money before by subscribing to groups and mags which have dropped dead the day after they sent their cheque. The only way you can hope to get anywhere is start small and hope that you can build up a good enough reputation for people to want to subscribe.

Start small

There's a couple of other reasons for starting small ... time and money! Running the NDUG takes me around sixty hours a week, minimum, which means goodbye to evenings out and weekends off for a start, and then you need to be able to cover a £500-per-year phone bill, the class of photocopier that you don't usually find under £2,000 even second hand, and a paper and postage bill that has to be seen to be belileved. If you have all this to spare, then please start a national group, and I promise to be one of your first members, but if you don't then please don't try to take on more than you can handle, it simply doesn't work. Exactly the same points apply to magazines. THere isn't quite as much work involved, but other costs are all very similar, and there's still the same problems involved.

Start small, and try to build up a reputation which will force you to expand. It isn't easy. You will find that all the people who told you waht a good idea it was and that they would be willing to help with the work will disappear like magic as soon as you try to pin them down to actually doing something at the time it needs doing, that the promised material, articles, etc. will never materialise, and that while everyone is all too willing to tell you what you should be doing, none of them ever want to assist in doing it! For what it is worth, the problem does improve with time, but of course you have to survive long enough to appreciate that!

Masochist

So why should you even attempt to start a group or a magazine? Well, possibly to gratify your repressed masochistic impulses, or perhaps because of your kind, generous and altruistic nature, or even because you want to keep interest in the Dragon alive, and possibly learn more about the old beast in the process.

So far as I'm concerned, running NDUG has taught me a lot I didn't know about the Dragon, got me some very useful contacts, a few very good friends, and even the odd enemy of two. It has written off any form of social like, and even with the help of some very good editors has given me far more work thatn I every imagined it could As an occupations it can be interesting, depressing, infuriating, and a right pain in the anatomy. It has been educational too, you'd be surprised at the adjectives I've invented!

Don't get the wrong idea, I'm most certainly not trying to put you off. What I AM trying to do is make sure you know what you'll be taking on, so that youy've a better than even chance of survival.

The point is that when you start something like a group or a magazine you're asking people to accept you on trust. You're asking them to pay you money, send you material, for something YOU have promised to run. YOu can't just take the money and then decide that it's all too much bother, too much work, and ditch the thing. That doesn't just leave a bunch of irate punters looking for you with their favourite piece of lead piping, or the local plods wanting you to assist them with the odd enquiry; those are minor points, and of interest only to yourself and your local casualty department. The real damage is done to the entire Dragon scene, people decide that enough is enough and they aren't going to risk the same thing happening again, so the geniune groups and magazines suffer, mail order software distributors (the only kind left) suffer, because YOU will have made people even more reluctant to risk their money. Get the picture?

Fun

Running a group or a magazine can be a lot of fun, and can help a lot of people, but please think the thing through properly before you start. We need more groups, especially local ones, the type which cover a town or county, because these are the ones which can dc most for the Dragon scene ... a really good network of local groups would be much better than one national one, even NDUG! and local Dragon magazines can carry much more of direct interest to their immediate readers than anational one, which has to take a more general view of things and of course misses out on a lot of local news, and of course, if you make a really good job of things, eventually you'll get subscribers from outside the immediate area, and then you're ready to 'go national' and leave the local scene for someone else to cover.

You, I KNOW I didn't do it that way, but that was four years ago, and the scene was much bigger then, and I had the advantage of being used to running a business, and could run a Group on the same lines. Times have changed now, the scene is smaller, and people are more reluctant to chance their money.

Anyway, let me know when you've got your group started and I'll give it a write-up, and that goes for your magazine as well, but please remember that I don't like writing obituaries!

Paul

Access and Search

***D. Hill** gets into his programs and out with the numbers*

HAVING recently spent many hours writing a long Basic program involving random access to disc files, I decided to change the name of one of the data files on the disc. I now had the time consuming task of searching through the entire program for all references to the old file name in order to change it to the new one.

The thought of all that searching prompted me to write a program to do it for me. My first thought was a short basic program but this created the problem of loading two basic program into memory which, although not a great problem was, nevertheless, a nuisance as well as taking up valuable memory.

What I wanted was a machine code program which would search a Basic program in memory for any string of characters entered and report the numbers of all lines containing that string. This program is the result.

The program is written in relocatable code, but is best loaded to 31900 after having reserved space for it. Typing

PCLEAR1:CLEAR50,31800

will leave the maximum amount of free memory to load the Basic program to be searched. A hex dump with checksum is included which can be loaded with Pam d'Arcy's hexloader (*Dragon User*, June 1985).

The program first checks that there is, in fact, a Basic program in memory, sets Flag B (output to printer-?) to 0, then asks for the string to be searched for.

Tokens

As I am sure most readers are aware, all Basic command words are stored as tokens, therefore, the next stage is to expand the line from its stored form in order to search it. If the string is found, you are asked whether the results are to go to a printer. If the answer is yes then Flag B is set to 1. The number of the line continuing the string is then displayed, the sub routine 'Device' checks Flag B and copies the results to the printer, if it's set.

The maximum length of the string is set at 20 and an error trap is included to ensure that this is not exceeded. The sub routine 'SCFULL' ensures that the results do not scroll off the screen until a key is pressed.

I'm sure that the more 'expert' programmers among you could improve upon my efforts but, nevertheless, I thought it could be of interest to others, if only as an example of what can be achieved by someone with no knowledge of computing other than that learned from the invaluable articles published in *Dragon User* and a copy of *Inside the Dragon*.

When first written, every time the program ended and returned to the basic command mode an SN Error was flagged. Apparently the computer was looking at the area of decrunched text and deciding that it was invalid word. The **clear imput buffer routine** is my solution to that problem. I'm not exactly sure how it works, but it does.

Help!

There are several articles in *Dragon User* and *Inside the Dragon* explaining how a Basic program is stored in memory but I couldn't find anything to explain exactly what happens when you type in something in direct mode and press enter. Maybe one of you knowledgeable people could enlighten me by writing an article on which locations are used etc.

Finally my sincere thanks to the many people who submit informative articles and to *Dragon User* for publishing them.

```
31000 BD,BA,77,6F,8D,02,AE,EC,9F,00,   CHKSUM=  1317
31010 19,10,83,00,00,26,07,30,8D,01,   CHKSUM=  407
31020 F1,8D,6B,39,BD,BA,77,30,8D,01,   CHKSUM=  1230
31030 74,8D,61,8D,6C,8D,3E,BD,BA,77,   CHKSUM=  1300
31040 17,00,A2,6D,8D,02,84,26,0D,6F,   CHKSUM=  731
31050 8D,02,80,30,8D,01,EA,8D,47,17,   CHKSUM=  930
31060 01,1B,86,0D,BD,80,0C,6F,8D,02,   CHKSUM=  758
31070 6E,30,8D,02,34,8D,35,BD,A0,EA,   CHKSUM=  1130
31080 81,59,27,C4,8E,02,DD,9F,A6,7F,   CHKSUM=  1270
31090 02,DD,7F,02,DE,7F,02,DF,39,BD,   CHKSUM=  1172
31100 BA,77,30,8D,01,E5,8D,16,BD,A0,   CHKSUM=  1236
31110 EA,81,53,27,0A,81,50,26,EC,6C,   CHKSUM=  1086
31120 8D,02,3A,20,04,6F,8D,02,34,39,   CHKSUM=  600
31130 A6,80,26,01,39,BD,80,0C,17,00,   CHKSUM=  742
31140 E3,20,F3,5F,30,8D,02,0B,34,10,   CHKSUM=  867
31150 BD,A0,EA,35,10,81,0D,27,1A,BD,   CHKSUM=  1048
31160 80,0C,81,08,26,07,30,1F,5A,2B,   CHKSUM=  534
31170 E4,20,E7,A7,80,5C,C1,14,23,E0,   CHKSUM=  1350
31180 8D,08,17,FF,5F,E7,8D,01,F7,39,   CHKSUM=  1199
31190 86,0D,BD,80,0C,30,8D,01,21,8D,   CHKSUM=  840
31200 B9,BD,A0,EA,39,6F,8D,01,E2,9E,   CHKSUM=  1462
31210 19,10,AE,84,10,8C,00,00,27,7C,   CHKSUM=  666
31220 34,20,EE,02,34,40,BD,8F,08,35,   CHKSUM=  833
31230 40,8E,02,DD,31,8D,01,B1,E6,8D,   CHKSUM=  1168
31240 01,C2,A6,80,4D,26,02,20,5A,A1,   CHKSUM=  889
31250 A4,27,02,20,EB,5A,27,04,31,21,   CHKSUM=  687
31260 20,EC,6D,8D,01,A9,26,15,86,0D,   CHKSUM=  894
31270 8D,60,30,8D,00,AA,17,FF,6B,8D,   CHKSUM=  1122
31280 40,86,0D,BD,80,0C,17,00,4F,DC,   CHKSUM=  862
```

```
31290 88,10,83,05,9F,25,02,8D,4F,1F,  CHKSUM=  737
31300 30,34,06,BD,95,7A,6D,8D,01,7F,  CHKSUM=  944
31310 27,0D,0A,6F,0A,6F,35,06,BD,95,  CHKSUM=  691
31320 7A,0F,6F,20,02,35,06,86,2C,BD,  CHKSUM=  708
31330 80,0C,17,00,21,6C,8D,01,60,35,  CHKSUM=  595
31340 10,16,FF,7B,39,31,8D,01,42,E6,  CHKSUM=  960
31350 8D,01,53,C1,00,26,01,39,5A,A6,  CHKSUM=  770
31360 A0,BD,80,0C,8D,02,20,F1,6D,8D,  CHKSUM=  1155
31370 01,41,2F,03,BD,80,0F,39,34,16,  CHKSUM=  579
31380 30,8C,79,6A,8D,01,32,17,FE,FC,  CHKSUM=  1136
31390 6C,8D,01,2B,BD,A0,EA,BD,BA,77,  CHKSUM=  1370
31400 35,16,39,45,4E,54,45,52,20,53,  CHKSUM=  629
31410 45,41,52,43,48,20,53,54,52,49,  CHKSUM=  709
31420 4E,47,20,41,4E,44,20,20,50,52,  CHKSUM=  618
31430 45,53,53,20,20,20,45,4E,54,45,  CHKSUM=  631
31440 52,20,20,20,20,00,54,48,45,20,  CHKSUM=  467
31450 46,4F,4C,4C,4F,57,49,4E,47,20,  CHKSUM=  721
31460 4C,49,4E,45,53,20,43,4F,4E,54,  CHKSUM=  719
31470 41,49,4E,20,54,48,45,20,53,54,  CHKSUM=  672
31480 52,49,4E,47,2E,2E,2E,00,53,54,  CHKSUM=  609
31490 52,49,4E,47,20,54,4F,4F,20,4C,  CHKSUM=  686
31500 4F,4E,47,20,50,52,45,53,53,20,  CHKSUM=  689
31510 41,4E,59,20,4B,45,59,00,4E,4F,  CHKSUM=  654
31520 20,42,41,53,49,43,20,50,52,4F,  CHKSUM=  659
31530 47,52,41,4D,4D,45,20,49,4E,20,  CHKSUM=  656
31540 4D,45,4D,4F,52,59,00,54,48,45,  CHKSUM=  698
31550 52,45,20,41,52,45,20,4E,4F,20,  CHKSUM=  620
31560 4F,43,43,55,52,52,45,4E,43,45,  CHKSUM=  745
31570 53,20,4F,46,20,54,48,45,20,53,  CHKSUM=  636
31580 54,52,49,4E,47,2E,2E,2E,2E,2E,  CHKSUM=  618
31590 00,20,20,52,45,53,55,4C,54,53,  CHKSUM=  626
31600 20,54,4F,20,53,43,52,45,45,4E,  CHKSUM=  675
31610 20,20,4F,52,20,50,52,49,4E,54,  CHKSUM=  654
31620 45,52,20,20,20,50,52,45,53,53,  CHKSUM=  644
31630 20,73,20,4F,52,20,70,20,00,41,  CHKSUM=  581
31640 4E,4F,54,48,45,52,20,53,45,41,  CHKSUM=  713
31650 52,43,48,3F,3F,20,20,50,52,45,  CHKSUM=  642
31660 53,53,20,79,20,4F,52,20,6E,00,  CHKSUM=  654
```

```
1389                  *SEARCH  A BASIC PROGRAMME FOR *
1389                  *ANY STRING INPUT.DEFAULT LOAD *
1389                  *  ADDRESS=31000.TYPE PCLEAR1  *
1389                  * CLEAR50,31000 BEFORE LOADING *
1389                  *    BY DENNIS HILL (1987)     *
1389                  ********************************
7918  7918                    ORG    31000
7918                          PUT    5000
7918  800C            OUTCH   EQU    $800C
7918  800F            OUTCHP  EQU    $800F
7918  A0EA            WAIT    EQU    $A0EA
7918  BA77            CLS     EQU    $BA77
7918  957A            OUTNUM  EQU    $957A
7918                  ********************************
7918 BDBA77                   JSR    CLS  ;CLS IN ROM
791B 6F8D02AD                 CLR    FLAGB,PCR
791F EC9F0019                 LDD    ($19)CHECK FOR
7923 10830000                 CMPD   £0 ;PROG.IN
7927 2607                     BNE    START  ;MEMORY
7929 308D01F1                 LEAX   NOMES,PCR
792D 8D6B                     BSR    OUTSTR
792F 39                       RTS
```

```
7930                   ********************************
7930                   *    PRINT PROMPT MESSAGE      *
7930                   *   AND INPUT SEARCH STRING    *
7930                   ********************************
7930 BDBA77            START  JSR   CLS
7933 308D0174                 LEAX  PROMPT,PCR
7937 8D61                     BSR   OUTSTR
7939 8D6C                     BSR   INSTR
793B 8D3E                     BSR   PRTER   ;PRINTER??
793D                   ********************************
793D                   * DECRUNCH,SEARCH,PRINT LINE NO*
793D                   ********************************
793D BDBA77                   JSR   CLS
7940 1700A2                   LBSR  GETLN
7943 6D8D0283                 TST   NU,PCR ;FOUND???
7947 260D                     BNE   RETURN
7949 6F8D027F                 CLR   FLAGB,PCR
794D 308D01EA                 LEAX  NOTFND,PCR
7951 8D47                     BSR   OUTSTR
7953 17011B                   LBSR  PRTSTR
7956                   ********************************
7956                   *      ANOTHER SEARCH??        *
7956                   ********************************
7956 860D              RETURN LDA   £$0D
7958 BD800C                   JSR   OUTCH
795B 6F8D026D                 CLR   FLAGB,PCR
795F 308D0234                 LEAX  MORMES,PCR
7963 8D35                     BSR   OUTSTR
7965 BDA0EA                   JSR   WAIT
7968 8159                     CMPA  £'Y
796A 27C4                     BEQ   START
796C                   ********************************
796C                   *CLEAR INPUT BUFFER ELSE ERROR *
796C                   *      ON RETURN TO BASIC      *
796C                   ********************************
796C 8E02DD                   LDX   £$2DD
796F 9FA6                     STX   $A6
7971 7F02DD                   CLR   $2DD
7974 7F02DE                   CLR   $2DE
7977 7F02DF                   CLR   $2DF
797A 39                       RTS
797B                   ********************************
797B                   *    OUTPUT TO PRINTER???
797B                   ********************************
797B BDBA77            PRTER  JSR   CLS
797E 308D01E5                 LEAX  DEVMES,PCR ;RESULTS
7982 8D16                     BSR   OUTSTR       ;TO
7984 BDA0EA                   JSR   WAIT      ;PRINTER??
7987 8153                     CMPA  £'S
7989 270A                     BEQ   NOPRT   ; NO.
798B 8150                     CMPA  £'P
798D 26EC                     BNE   PRTER   ;INVALID
798F 6C8D0239                 INC   FLAGB,PCR
7993 2004                     BRA   YESPRT
7995 6F8D0233          NOPRT  CLR   FLAGB,PCR
7999 39                YESPRT RTS
799A                   ********************************
799A                   *     PRINT A TEXT STRING      *
799A                   ********************************
799A A680              OUTSTR LDA   ,X+
799C 2601                     BNE   OUT1
799E 39                       RTS
```

```
799F BD800C      OUT1   JSR   OUTCH
79A2 1700E3             LBSR  DEVICE  ;PRINTER???
79A5 20F3               BRA   OUTSTR
79A7                    ********************************
79A7                    *INPUT STRING TO SEARCH FOR AND*
79A7                    *  STORE LENGTH IN LENSTR      *
79A7                    ********************************
79A7 5F          INSTR  CLRB
79A8 308D020A           LEAX  STRING,PCR
79AC 3410        GETCH  PSHS  X
79AE BDA0EA             JSR   WAIT ;KEY PRESSED??
79B1 3510               PULS  X     ;YES
79B3 810D               CMPA  £$0D  ;ENTER???
79B5 271A               BEQ   FIN   ;YES
79B7 BD800C             JSR   OUTCH ;NO PRINT IT
79BA 8108               CMPA  £08   ;BACKSPACE??
79BC 2607               BNE   STORE ;NO
79BE 301F               LEAX  -1,X  ;YES.GO BACK
79C0 5A                 DECB        ;ONE IF NOT
79C1 2BE4               BMI   INSTR ;NEG VALUE
79C3 20E7               BRA   GETCH ;TRY AGAIN
79C5                    *****STORE SEARCH STRING********
79C5 A780        STORE  STA   ,X+
79C7 5C                 INCB
79C8 C114               CMPB  £20  ;MAX LENGTH
79CA 23E0               BLS   GETCH  ;OK
79CC 8D08               BSR   TOOBIG ;TOO LONG
79CE 17FF5F             LBSR  START
79D1 E78D01F6    FIN    STB   LENSTR,PCR
79D5 39                 RTS
79D6                    ********************************
79D6                    *     STRING TOO LONG          *
79D6                    ********************************
79D6 860D        TOOBIG LDA   £$0D
79D8 BD800C             JSR   OUTCH
79DB 308D0121           LEAX  ERROR,PCR
79DF 8DB9               BSR   OUTSTR
79E1 BDA0EA      BIG1   JSR   WAIT
79E4 39                 RTS
79E5                    ********************************
79E5                    * DECRUNCH LINE, SEARCH IT AND *
79E5                    *  PRINT LINE NUMBER IF FOUND  *
79E5                    ********************************
79E5 6F8D01E1    GETLN  CLR   NU,PCR  ;LINES FND.
79E9 9E19               LDX   $19 ;STARTOF PROG.
79EB 10AE84      NXTLN  LDY   ,X     ;NXT LINE ADD
79EE 108C0000           CMPY  £0     ;END OF PROG?
79F2 277C               BEQ   END    ;YES
79F4 3420               PSHS  Y      ;SAVE NXT ADD
79F6 EE02               LDU   2,X    ;GET LINE NO.
79F8 3440               PSHS  U      ;SAVE IT
79FA BD8F08             JSR   $8F08  ;DECRUNCH
79FD 3540               PULS  U      ;RECOVER
79FF                    ****POINT TO DECRUNCHED TEXT****
79FF 8E02DD             LDX   £$02DD
7A02 318D01B0    RESET  LEAY  STRING,PCR
7A06 E68D01C1           LDB   LENSTR,PCR
7A0A A680        NXTCHR LDA   ,X+
7A0C 4D                 TSTA  ;END OF LINE?
7A0D 2602               BNE   SEARCH ;NO
7A0F 205A               BRA   NXTLN2 ;YES
```

```
7A11            ********************************
7A11            *   SEARCH LINE FOR STRING     *
7A11            ********************************
7A11 A1A4       SEARCH CMPA  ,Y   ;CHARACTER
7A13 2702              BEQ   MATCH    ;MATCHED??
7A15 20EB              BRA   RESET    ; NO
7A17 5A         MATCH  DECB           ;YES
7A18 2704              BEQ   PRTMES   ;ALL MATCH
7A1A 3121              LEAY  1,Y      ;NO
7A1C 20EC              BRA   NXTCHR   ;TRY AGAIN
7A1E            ********************************
7A1E            * PRINT LINE NUMBER IF MATCHED *
7A1E            ********************************
7A1E 6D8D01A8   PRTMES TST   NU,PCR ;1ST ONE SO
7A22 2615              BNE   CHKSC  ;ENDMESSAGE
7A24 860D              LDA   £$0D
7A26 8D60              BSR   DEVICE
7A28 308D00AA          LEAX  ENDMES,PCR
7A2C 17FF6B            LBSR  OUTSTR
7A2F 8D40              BSR   PRTSTR
7A31            *****START ON NEW LINE**********
7A31 860D              LDA   £$0D
7A33 BD800C            JSR   OUTCH
7A36 17004F            LBSR  DEVICE
7A39            ********************************
7A39 DC88       CHKSC  LDD   $88      ;SCREEN
7A3B 1083059F          CMPD  £$59F  ;FULL??
7A3F 2502              BLO   PRTLN  ;NO
7A41 8D4F              BSR   SCFULL ;YES
7A43 1F30       PRTLN  TFR   U,D
7A45 3406              PSHS  D
7A47 BD957A            JSR   OUTNUM
7A4A 6D8D017E          TST   FLAGB,PCR ;OUT TO
7A4E 270D              BEQ   NOCOPY   ;PRINTER?
7A50 0A6F              DEC   $6F      ;YES,SET
7A52 0A6F              DEC   $6F      ;TO -2
7A54 3506              PULS  D
7A56 BD957A            JSR   OUTNUM
7A59 0F6F              CLR   $6F          ;SET TO 0
7A5B 2002              BRA   COMMA
7A5D 3506       NOCOPY PULS  D
7A5F 862C       COMMA  LDA   £', ;PRINT A ,
7A61 BD800C            JSR   OUTCH
7A64 170021            LBSR  DEVICE
7A67 6C8D015F   INC    INC   NU,PCR
7A6B 3510       NXTLN2 PULS  X ;ADD OF NXT LINE
7A6D 16FF7B            LBRA  NXTLN
7A70 39         END    RTS   ;END OF BASIC PROG.
7A71            ********************************
7A71            *    PRINT SEARCH STRING       *
7A71            ********************************
7A71 318D0141   PRTSTR LEAY  STRING,PCR
7A75 E68D0152          LDB   LENSTR,PCR
7A79 C100       PRINT1 CMPB  £0  ;PRINT THE
7A7B 2601              BNE   DEC  ;SEARCH
7A7D 39                RTS        ;STRING
7A7E 5A         DEC    DECB
7A7F A6A0              LDA   ,Y+
7A81 BD800C            JSR   OUTCH
7A84 8D02              BSR   DEVICE   ;PRINTER??
7A86 20F1              BRA   PRINT1
```

```
7A88                 *********************************
7A88                 *OUTPUT TO PRINTER IF FLAG SET**
7A88                 *********************************
7A88 6D8D0140   DEVICE TST   FLAGB,PCR
7A8C 2F03              BLE   DEVEND
7A8E BD800F            JSR   OUTCHP
7A91 39         DEVEND RTS
7A92                 *********************************
7A92                 * SCREEN FULL.SO PRINT MESSAGE *
7A92                 *    AND WAIT FOR KEY PRESS     *
7A92                 *********************************
7A92 3416       SCFULL PSHS  X,D
7A94 308C79            LEAX  SCMESS,PCR
7A97 6A8D0131          DEC   FLAGB,PCR ;PRINTER
7A9B 17FEFC            LBSR  OUTSTR   ;OFF AND
7A9E 6C8D012A          INC   FLAGB,PCR  ;ON
7AA2 BDA0EA            JSR   WAIT
7AA5 BDBA77            JSR   CLS
7AA8 3516              PULS  X,D
7AAA 39                RTS
7AAB                 *********************************
7AAB 454E544552 PROMPT FCC /ENTER SEARCH STRING/
7ABE 20414E4420        FCC / AND  PRESS   ENTER/
7AD1 2020202000        FCC /    /,0
7AD6 5448452046 ENDMES FCC /THE FOLLOWING LINES/
7AE9 20434F4E54        FCC / CONTAIN THE STRING/
7AFC 2E2E2E00          FCC /.../,0
7B00 535452494E ERROR  FCC /STRING TOO LONG /
7B10 5052455353 SCMESS FCC /PRESS ANY KEY/,0
7B1E 4E4F204241 NOMES  FCC /NO BASIC PROGRAMME/
7B30 20494E204D        FCC / IN MEMORY/,0
7B3B 5448455245 NOTFND FCC /THERE ARE NO OCCUR/
7B4D 52454E4345        FCC /RENCES OF THE STRIN/
7B60 472E2E2E2E        FCC /G...../,0
7B67 2020524553 DEVMES FCC /  RESULTS TO SCREEN/
7B7A 20204F5220        FCC /  OR PRINTER   PRES/
7B8D 532073204F        FCC /S s OR p /,0
7B97 414E4F5448 MORMES FCC /ANOTHER SEARCH??/
7BA7 2020505245        FCC /  PRESS y OR n/,0
7BB6            STRING RMB  20
7BCA            NU     RMB  1
7BCB            LENSTR RMB  1
7BCC            FLAGB  RMB  1
7BCD
```

Winners and Losers

Every month Gordon Lee will look at some prize programming

WHEN the American research physicist Frank Gray first developed the code series which now bears his name, he could have had no idea of the problems which it would cause our competitors nearly half a century later!

Regular entrant to the competition D J Gray (note that name) thought he had cracked it by contacting his Uncle Frank, only to be told that "it wasn't him and he had only been to American once and could I please not tell Aunty Betty". Sorry, D J, I hope that hasn't rattled too many family skeletons.

However, undaunted, D J, along with a fair few others, managed to crack the code to produce a Gray to binary converter. And what a selection of entries there were, including a number of multi-page treatises using techniques redolent of Exclusive-OR gates, flip flops (which I always thought were a type of footwear), Boolean functions, and Karnaugh maps. If all this seems a little daunting then for pure simplicity let's turn jto Fred Taylor of Middlesbrough, whose 'bare bones' program is given here (**listing one**). Although not error trapped (Fred did include a longer error-trapped program) it is wonderfully straightforward and does the job using the same algorithm as that outlines on the Answer page of the June issue.

Listing two is an even shorter program using this same algorithm, and has the added advantage that it can be used to convert Gray codes of infinite length! I'll leave it to interested readers to work out the logic behind this listing.

I'm indebted to Fred Willers of Stone in Staffordshire for some additional notes on evaluating a Gray code directly into its decimal equivalent. Each Gray code 'bit' which is set to 1 is evaluated using the formula $(2^N)-1$ where N is the bit position (counting from the right-hand end). Each of the values of the bits set to 1 are then alternately added and subtracted. For example, taking the Gray code 10110011001 we would get:

N	10	9	8	7	6	5	4	3	2	1
$(2^N)-1$	1023	511	255	127	63	31	15	7	3	1
Gray Code	1	0	1	1	0	0	1	1	0	1
Decimal	1023		-255	+127			-15	+7		-1

Decimal equivalent = 886

While on the subject of counting from the right hand end, I am taken to task by Randy Longshore of Chesterfield, over the answer to the September '87 competition. This was the calculation of a 34537-digit number or at least the digits at positions twenty-thousand and one to twenty-thousand and ten. "Why", he asks, "are the digits counted from the left to the right?" and not, as Randy had done, from the right (decimal point) end. I suppose that the logical answer would be that, had the answer been, say, 471397021, and I had asked for the fourth digit, the answer would have been 3. Consequently, the fact that the number has many thousands of digits will not affect the end from which you count. QED.

I can afford to be flippant without fear of a punch on the nose from Randy, as he hails from Chesterfield USA and not Chesterfield UK! Finally, on the subject of the tie-breakers (and I'm still trying to work out Robin Telkman's), my personal favourite comes from D J Gray.

"There is no doubt that I am the most eligible person to send to a paradise island. I have my own eight records and solar powered record player. I have no need of the complete works of Shakespeare and am prepared to borrow the family bible. My Dragon would of course be the luxury item, so all I can ask for is a regular delivery of *Dragon User*. (*Aaaah! Ed.*).

As a slightly harrassed compiler of competition problems, I too would put forward a strong claim for eligibility. Like D J Gray, I too would happily forgo the works of Shakespeare, but would much prefer to dig up the man himself, if only to throw stones at him! (*And on that enigmatic note ...*).

Listing 1

```
10 INPUT"GRAY CODE";G$
20 FOR X=1 TO LEN(G$)
30 D$=MID$(G$,X,1):IF X<2 THEN 50
40 IF D$=RIGHT$(B$,1) THEN D$="0" ELSE D$="1"
50 B$=B$+D$:NEXT X
60 PRINT"BINARY CODE";B$
```

Listing 2

```
10 Z=0
20 D$=INKEY$(0):IF D$=""THEN 20
30 D=VAL(D$):Z=ABS(D<>Z):PRINTSTR$(Z);:GOTO 20
```

Vive les differences

***Graham Smith** compares the Dragons 64 and 32*

WITH so may Dragon 64s becoming available on the second hand market at reasonable prices (in the region of £50 to £75), I am sure that many people will be considering if it is worth buying one. Before I describe the main differences between the two Dragons, I will just cover a few of the possible reasons for buying a 64.

If this will be your first Dragon (where have you been?), and you only intend to play commercially available games (and be honest here, nobody still tries to justify a computer by saying they want to do their home accounts on it), 50 quid is not much to outlay for a good machine, but almost all the games you will buy will also run on a 32 which can often be picked up for half the price and you will never use the extra facilities available on the 64. Having said that, if you can afford the extra few bob to buy the 64, you may thank yourself later when your interests widen out.

If you have a 32 and fancy a 64, remember you won't get much for the 32 if you try to sell it, especially if you are one of those people who paid the full £179 when they were new. I would advise you to keep it for backup or spares, as many of the bits, such as the keyboard or power supply, can be swapped over, and they would cost more to replace than you would get for the old 32.

The built-in RS232 port on the 64 can be enough to justify buying for anyone interested in electronic mail systems (such as bulletin boards) as adding an RS232 port to a 32 usually wastes the cartridge port. If you are interested in an example, the Maplin Elecltronics RS232 self-assembly kit for a Dragon 32 costs just about £14, has no case, and fits in the cartridge port.

Anyone wanting to use one of the alternative operting systems (FLEX, OS-9 or BASIC2) must have a Dragon 64 (or a suitably upgraded 32), and a disc drive.

Now to describe the main differences between the two machines. Externally, the 64 is grey rather than cream, the label says Dragon 64 and on the side by the joystick ports is another socket marked S.I/O. which is the RS232 port. Internally, the circuit board is similar to the 32 with a few extra chips squeezed in. These are the extra ram chips and a rom holding the reassembled Basic interpreter. (I will come back to this rom later).

There are a few minor differences between the Basic on the 32 and the 64. Two new commands DLOAD and DLOADM are similar to the CLOAD and CLOADM but are directed at the S.I/O. port for transfering files from another computer. This port can also be used with a serial printer if you POKE &H3FF,1 and then PRINT-2 or

Continued on page 19

GOSUB

***Paul Burgin** captures screens the simple way.*

HAVING completed or being about to write a programming masterpiece, it's always a great improvement to include well presented and colourful tables, menus or title screens. However, working out CHR numbers, PRINT @ positions and centering is often quite a bind, especially where graphics are concerned, so here's a program to cut out this hard work.

Yes, this is actually a program which writes Basic! After all, why spend hours producing something which the computer can do in less than fifteen seconds?

The program is written in machine code and is very simple to use. Simply load a Basic program and give *Gosub Writer* a screen, and it will write a loop onto the end of your program, which can be called using GOSUB.

When the program runs, it scans the screen from top to bottom, coding it as it scans. The first line of the gosub is always a REM statement and followed by a CLS. The lines are numbered in tens starting from the next multiple of 100. The routines end with a RETURN. Nothing escapes the eye of the computer, so that when the routines are called, it makes an exact replica of what was on the screen.

Most of the lines will be PRINT statements, which contain direct text, or for multiple characters, the computer will choose to use the STRING$ command. The program will also code graphics using the CHR$ command, and characters which cannot be printed will be POKEd to the screen. For position, the computer will use the PRINT@ command where necessary and to ensure that the screen doesn't scroll, the last space is always POKEd if used.

The length of the lines are also kept under control, with the computer starting a new line for every screen line and extras when too many statements build up.

Entering the program

First type **listing one**, the Basic hexloader and run it. You must now enter all the data for the machine code held in **listing two**. When you first start, enter 1536 for the start address, then enter each line of data digits, followed by the checksum. Any lines containing errors will need to be entered again. If you need to break off before typing the whole listing, press BREAK and save using:

CSAVEM "GWRITER", 1536, S, 0

You can then continue at any time by reloading the hexloader and machine code using CLOADM. Don't forget to note the location from which you have to carry on.

When you have typed the whole listing (location = 3036) press BREAK. If you have a Dragon 64, you must do the following patch;

POKE 2965, 1:POKE 2977,1:POKE 2997,1:POKE 3004,2

To save *Gosub writer*, type;

CSAVEM"GWRITER",1536,3030,1536

The program is now ready to use, but it's unlikely that you can do much with it unless you have a screen editor to create the screens in the first place. For those who do not, I provide one here, but if you do have one, and can liberate it to being co-resident with *Gosub Writer* and a program, you will need to know the following. The entry address for *Gosub Writer* is 1536. The screen to be coded is assumed to be positioned from 1024 to 1535 and locations 25 and 26 must point to the end of Basic; locations 27 and 28 must equal locations 25 and 26, plus 2, if there's no Basic.

To load a screen manually from a tape for testing *Gosub Writer* use;

POKE 104,0:CLOADM:EXEC 1536

This poke allows loading without the cursor flashing to corrupt the screen.

To enter the screen editor use the hexloader to enter the short patch, *listing three*, then as direct commands type;

POKE 25,14:POKE26,74:POKE3621,74:POKE3643,74:NEW

Next, type in **listing four** very carefully. DO NOT RUN IT YET. When typed, you can save *Gosub Writer*, the patch and Basic screen editor by typing;

CSAVEM"GWRITER",1536,6143,3614

Now that everything is installed you can take advantage of the features of *Gosub Writer*. Using the program is simple.

1) CLOAD the receiving Basic program, if any. Do not PCLEAR below PCLEAR 3.
2) Load *Gosub Writer* using CLOADM:EXEC

Don't worry that there are two Basic programs in the computer. The short patch takes care of this.

The screen editor is menu driven and supports cassette facilities. The keys for editing are as follows:

ENTER — Swap between text/graphics mode
Arrows — Move cursor and draw

Arrows + shift — Move cursor
0 to 8 — Colours for graphics
CLEAR — Return to menu
SHIFT + 0 — Swap between upper/lower case (even for numbers)

If all this typing is too daunting, I will be happy to supply the complete program and editor on tape for £3.50. Payable to Paul Burgin, 18 Moorcroft Road, Sheffield S10.

Listing 1

```
10 'HEXLOADER - Enter each line of
20 'digits and then input the checksum
30 PCLEAR2:CLS:INPUT"START FROM";S
40 PRINTCHR$(8)S;:LINEINPUT":";A$:CH=0
50 FORX=1TO24STEP2:K=VAL("&H"+MID$(A$,X,
2)):CH=CH+K:POKES,K:S=S+1:NEXT
60 PRINT" =";:INPUTCS:IFCS<>CH THENSOUND
20,5:S=S-12:PRINT"ERROR - ENTER"S"AGAIN"
70 GOTO40
```

Listing 2

```
1536:8E0BD76F808C0BFC23F9BD0B= 1494
1548:B4DC1B931910830002220E9E= 954
1560:19BF0BE28E0064BF0BDD7E06= 1250
1572:6F9E19BF0BE4AE9F0BE4BF0B= 1498
1584:E2AE9F0BE2BF0BE6BE0BE68C= 1799
1596:00002708BE0BE2BF0BE420E2= 1162
1608:BE0BE4AE02BF0BDD8E0064BF= 1461
1620:0BDBBE0BDDBC0BDB250BBE0B= 1319
1632:DB308864BF0BDB20EDBE0BDB= 1613
1644:BF0BDD12BD076E8683BD079F= 1367
1656:BD07A9BD076E86A0BD079FBD= 1509
1668:07A912B605FFB70BE8816027= 1326
1680:058620B705FF7F0BF58E0400= 1143
1692:9F888601B70BEDB70BF68E04= 1447
1704:00BF0BE9BF0BEBB60BED8101= 1432
1716:2702200B7F0BEDBD076E8687= 1034
1728:BD079FBD07C4B60BEDB70BF6= 1617
1740:7C0BF58E06D7BD90E5201020= 1385
1752:2020202020202020203C4C494E= 543
1764:452000F60BF54FBD957A7D0B= 1278
```

```
1776:ED261C8E06FBBD90E5201A20= 1354
1788:20424C414E4B3E0D00202043= 598
1800:4F4445443E0D008E0705BD90= 846
1812:E5B60BF5C60A3DD78CC601BD= 1679
1824:BAA0B60BED81012603BD07A9= 1312
1836:BE0BEB308820BF0BEBBF0BE9= 1524
1848:8C05E01023FF70127D0BED26= 1216
1860:08863BBD079FBD07A912BD07= 1135
1872:6E8691BD079FBD07A9BE0BE2= 1536
1884:6F806F806F84BE0BE230029F= 1357
1896:1B9F1D9F1F3912FC0BE2C301= 1165
1908:2C10FF0BFC10B30BFC250D9E= 1244
1920:1B301E6F806F84C60C7E8344= 1122
1932:10BE0BDDBE0BE210AF023004= 1110
1944:BF0BF77F0BF93912BE0BF7A7= 1526
1956:80BF0BF73912BE0BF76F80BF= 1530
1968:0BF710BE0BE2AFA4BF0BE2BE= 1658
1980:0BDD300ABF0BDD3912BE0BE9= 1222
1992:BF0BEEA69F0BEEB70BFAB60B= 1651
2004:FAA19F0BEE2667BE0BEE3001= 1448
2016:BF0BEEBE0BE9308820BF0BFC= 1544
2028:BE0BEEBC0BFC25DEB60BFA81= 1721
2040:60260139B60BFA8180253FBE= 1182
2052:0BE9BF0BD7BD0BC186FFBD07= 1639
2064:9F86A0BD079F8628BD079F8E= 1479
2076:0020BF0BDFBD0B8D862CBD07= 1172
2088:9FF60BFA4FFD0BDFBD0B8D86= 1707
2100:01B70BED8629BD079F863B7E= 1281
2112:079F8601B70BEDBE0BE9BF0B= 1368
2124:D7A69F0BD78160260ABE0BD7= 1455
2136:3001BF0BD720EEBE0BE9BC0B= 1369
2148:D72607B60BF681012703BD0B= 1071
2160:C1A69F0BD7B70BE17C0BF9B6= 1729
2172:0BF98112260BBD07A9BD076E= 1127
2184:8687BD079FB60BE181801025= 1352
2196:00E3FC0BD7B30BE91083001D= 1304
2208:102200ACBE0BD7B60BE1A101= 1218
2220:102600A0A1021026009AA103= 749
2232:102600948601B70BFBBE0BD7= 1198
2244:3001BF0BD9A69F0BD9B10BE1= 1434
2256:26487C0BFBBE0BD93001BF0B= 1165
2268:D9BE0BE9308820BC0BD926E1= 1546
2280:86FFBD079F86A0BD079F8628= 1567
2292:BD079FF60BFB4FFD0BDFBD0B= 1629
2304:8D862CBD079FF60BE14FFD0B= 1499
2316:DFBD0B8D8629BD079F863B7E= 1413
2328:079F86FFBD079F86A0BD079F= 1559
2340:8628BD079FF60BFB4FFD0BDF= 1603
2352:BD0B8D862CBD079FF60BE14F= 1435
2364:FD0BDFBD0B8D8629BD079FBE= 1548
2376:0BD9BF0BD77E0AF7BE0BD730= 1492
2388:01BF0BD786FFBD079F8691BD= 1630
2400:079F8628BD079FF60BE14FFD= 1509
2412:0BDFBD0B8D8629BD079F7E0A= 1241
2424:F78622BD079F7F0BF0B60BE1= 1566
2436:815F23058040B70BF0B60BE1= 1308
2448:812024058B60B70BF0B60BF9= 1313
2460:810E251A8622BD079F863BBD= 1111
2472:079FBD07A9BD076E8687BD07= 1302
2484:9F8622BD079FB60BE1811F23= 1295
2496:068140102500A57D0BF02606= 837
2508:B60BE1B70BF0B60BF0812226= 1486
2520:31BD079F863BBD079F86FFBD= 1530
2532:079F8691BD079F8628BD079F= 1329
2544:8633BD079F8634BD079F8629= 1256
2556:BD079F863BBD079F7C0BF97C= 1411
2568:0BF9B60BF0BD079FBE0BD730= 1512
2580:01BF0BD7BE0BE9308820BC0B= 1267
2592:D7260A8622BD079F863B7E07= 1112
2604:9FA69F0BD7B70BE181602706= 1399
2616:81801025FF40B60BE1816026= 1310
2628:11BE0BD7A601816026088622= 1039
2640:BD079F7E0AF7B60BE1816010= 1397
2652:27FF1F8622BD079F863BBD07= 1237
2664:9F7E0896BE0BD78C05FF274B= 1373
2676:8620BD079F8622BD079F863B= 1237
2688:BD079F863ABD079F8693BD07= 1379
2700:9FBE0BD7BF0BDFBD0B8D862C= 1519
2712:BD079FF60BE14FFD0BDFBD0B= 1603
2724:8D863ABD079F8687BD079F86= 1446
2736:22BD079FB60BF98B03B70BF9= 1416
2748:7E0A108622BD079F863BBD07= 1064
2760:9F863ABD079F8693BD079F86= 1476
2772:31BD079F8635BD079F8633BD= 1320
2784:079F8635BD079F862CBD079F= 1241
2796:F60BE84FFD0BDFBD0B8D39BE= 1643
2808:0BE9308820BC0BD72605863B= 1110
2820:7E079FA69F0BD78160270886= 1249
2832:3BBD079F7E0871BE0BD7BF0B= 1279
2844:F1A69F0BF181602614BE0BF1= 1543
2856:3001BF0BF1BE0BE9308820BC= 1330
2868:0BF126E539BE0BD7BF0BF3A6= 1603
2880:9F0BD78160260ABE0BD73001= 1123
2892:BF0BD720EE863BBD079F86FF= 1624
2904:BD079F86A0BD079F8628BD07= 1374
2916:9FFC0BD7B30BF3FD0BDFBD0B= 1757
2928:8D862CBD079F8633BD079F86= 1348
2940:32BD079F8629BD079F863BBD= 1317
2952:079F7E0871FC0BDF9E883410= 1261
2964:8E03F09F88BD957A35109F88= 1504
2976:8E03F0A6808160270B804034= 1198
2988:10BD079F351020EF8E03F086= 1230
3000:60A7808C040026F939128640= 1095
3012:BD079FFC0BD7830400FD0BDF= 1455
3024:8DBB862C7E079F0000000000= 798
```

Listing 3

```
3584:8E0400A680A78907FF8C0600= 1152
3596:26F5398E0400A6890800A780= 1092
3608:8C060026F5399E19BF0E478E= 1087
3620:0E499F19BD841FBD83ED7E84= 1438
3632:9FBE0E479F19BD06008E0E49= 1042
3644:9F1939BE0E479F197E837100= 1070
```

Listing 4

```
10 CLS:GOSUB330:A$="12345":GOSUB320:ONP
GOTO20,40,280,290,300
20 CLS:PRINT@128,0;STRING$(27,128),1;TAB
(8);"GREEN (GRAPHICS)":FORI=159TO255STEP
16:PRINT(I-127)/16;STRING$(27,I):NEXT:PR
INT9;TAB(11);"GREEN (TEXT)":PRINT@34,"EN
TER BACKGROUND COLOUR 0-9?":A$="01234567
89":GOSUB320:IFP=10THENCLSELSECLSP-1
```

```
30 EXEC&HE00:TL=1264:X=0:Y=0:C=P-1:IFC=9
THENC=0
40 IFTL=0THENTL=1264:C=0:X=0:Y=0
50 EXEC&HE0F
60 PK=PEEK(TL):POKE136,INT(TL/256):POKE1
37,TL-(PEEK(136)*256):POKE143,1:IFPK=128
THENPOKETL,255
70 J$=INKEY$:IFJ$=""THENEXEC32777:GOTO70
80 IFASC(J$)=9THENTN=TL+1ELSEIFASC(J$)=8
THENTN=TL-1ELSEIFASC(J$)=94THENTN=TL-32E
LSEIFASC(J$)=10THENTN=TL+32ELSETN=0
90 TA=(TL)AND(31):IF(ASC(J$)=9)AND(TA=31
)THENTN=TL-31ELSEIF(ASC(J$)=8)AND(TA=0)T
HENTN=TL+31
100 POKETL,PK:IFTN=0THEN120ELSETL=TN:IFT
L<1024THENTL=TL+512ELSEIFTL>1535THENTL=T
L-512
110 GOTO60
120 HU=ASC(J$):IFHU=12THEN270ELSEIFHU=13
THEN160ELSEIFHU>95THENHU=HU-32
130 IFHU>63THEN PK=HU ELSE PK=HU+64
140 IFPEEK(329)=0THENPK=PK-64
150 J$=CHR$(9):GOTO80
160 LC=1024+(INT(Y/2)*32)+INT(X/2)
170 JJ=PEEK(LC)
180 D=POINT(X,Y):IFD<0THEND=8
190 FORW=1TO3:SET(X,Y,C):SET(X,Y,D):NEXT
:RESET(X,Y)
200 V$=INKEY$:IFV$=""THEN180ELSEPOKELC,J
J:V=ASC(V$)
210 IFV=12THEN270ELSEIFV=13THEN60
220 IF(V=9)OR(V=93)THENX=X+1ELSEIF(V=8)O
R(V=21)THENX=X-1ELSEIF(V=94)OR(V=95)THEN
Y=Y-1ELSEIF(V=10)OR(V=91)THENY=Y+1
230 IF(V>47)AND(V<57)THENC=V-48:GOTO180
240 X=(X)AND(63):Y=(Y)AND(31)
250 IF(V=9)OR(V=8)OR(V=94)OR(V=10)THENRE
SET(X,Y):IFC>0THENSET(X,Y,C)
260 GOTO160
270 EXEC&HE00:GOTO10
280 CLS:PRINT@32,"SAVE SCREEN":PRINTSTRI
NG$(11,"-"):PRINT@128,;:LINEINPUT"FILENA
ME:";F$:EXEC&HE0F:CSAVEMF$,1024,1535,336
49:GOTO10
290 CLS0:CLOADM:EXEC&HE00:GOTO10
300 EXEC&HE0F:EXEC&HE31:CLS:INPUT"ANY MO
RE ? Y/N";N$:IFN$="Y"THEN10
310 EXEC&HE3F
320 X$=INKEY$:IFX$=""THEN320ELSEP=INSTR(
1,A$,X$):IFP=0THEN320ELSERETURN
330 PRINT@33,CHR$(129);STRING$(28,131);C
HR$(130)
340 PRINT@65,CHR$(133);"GOSUB WRITER BY
PAUL BURGIN.";CHR$(138)
350 PRINT@97,CHR$(132);STRING$(28,140);C
HR$(136)
360 PRINT@164,"1.";STRING$(2,32);"CREATE
 NEW SCREEN"
370 PRINT@196,"2.";STRING$(2,32);"EDIT C
URRENT SCREEN"
380 PRINT@228,"3.";STRING$(2,32);"SAVE S
CREEN TO TAPE"
390 PRINT@260,"4.";STRING$(2,32);"LOAD S
CREEN FROM TAPE"
400 PRINT@292,"5.";STRING$(2,32);"CODE A
ND MERGE SCREEN"
410 PRINT@361,"PRESS KEYS 1-5":RETURN
```

Continued from page 16

LLIST. The other well-known difference between 32 and 64 Basic is the bug with the USR call, which on the 32 required all calls to be two digit eg USR01 or USR02. This has been cured on the 64 which means some Basic games require modification before they are compatible.

The major difference between the 32 and 64 is the availability of the full 64K of ram. When you switch on your 64, you will find it identical to a 32 (other than the abovementioned minor changes). If you type PRINT MEM, you will get the same value as you would with a 32, (24871). If you now type EXEC, the 64 executes a short boot strap routine which switches it into 64K ram mode and copies the reassembled Basic rom mentioned above into the area normally reserved for the cartridge port. PRINT MEM now gives you 41241 and if you want even more, you can POKE 25,6:NEW to delete all the graphics pages. You will now have 47385 bytes to play with. This stage is often refered to as the 48K system, although the Basic interpreter now resides in ram above the 48K mark and can be corrupted, (or altered) by suitable POKEing if desired.

The alternative operating systems such as OS-9 and FLEX use the 64K ram facility of the Dragon 64 to replace the resident operating system. Many other utilities have been produced to utilise the extra memory available. *BASIC42* by Harris Micro Software modifies the existing Basic in ram and adds many additional facilities.

One advantage of the 64 that is not often mentioned, is the fact that many cartridge based programs can be saved to tape or disc and then reloaded (with the aid of a suitable m/c utility), into the original location without the need to insert a cartridge, thus saving wear on the cartridge port.

Communications

Write down your problem on the coupon below (make it as brief and legible as possible) together with your name and address and send it to Communication, 49 Alexandra Road, Hounslow, Middx TW3 4HP.

Problem ....

....

....

....

Name ....

Address ....

....

....

Display Switcher

***Ken Smith** devises a cheap hardware screen inverter.*

REGULAR readers will know that for some time, dragoneers have been complaining about the poor display they get when they connect their treasured machines to a monochrome monitor. When I purchased a monitor, I found this disappointment was wholly justified. You buy a monitor to improve the display, but what you get is considerably worse than with a black and white television. I have mentioned before in these pages that some software (*Basic42* for instance) which simulates a white screen gives an acceptable display. However, most programs which require a good display, such as word processors, use a black on green display which is hideous and barely readable. It seems that the purer signal exaggerates the poor picture quality. One reader did suggest that the Dragon's monitor output was for colour only and this, combined with my own experience, set me thinking.

Not being an electronics expert, I was looking for a software switch to change my green screens to white and was meeting with very little success. During this search it was suggested that I might be better off trying to do the job with hardware. What follows, is the result of my investigations.

Two cures

There are two ways of curing this problem, both of which involve disposing of the colour.

The first cure is to remove the chip marked LM1889 which is the unit that controls the colour circuits. On some machines, this is a plug in chip and so can be easily removed or replaced. However, on others it is soldered in and unless you are capable with a soldering iron, it is best not attempted. My machine falls into the latter category and the whole idea seemed some what drastic, so I abandoned it.

The second option is to switch off the colour crystal. This can be done relatively easily and can be made switchable. This second option seemed far more promising and after months of thinking about it (my soldering is lousy) this was the course I embarked on. The materials required are listed below. Total cost is about £1.50p and the job takes about fifteen minutes. The materials required are one miniature single pole toggle switch, one 0.1 micro farad capacitor of any voltage over 16V, and half a metre of 0.6 mm insulated stranded copper wire.

Switch

The first task is to find a suitable site for your switch and drill a 5mm hole to mount it. I chose a position just under the left side of the keyboard, making it possible to switch from colour to monochrome without straying far from the keys.

Next, solder one of the capacitor tails to one of the switch terminals. Locate the crystal; this is a small metal can, and is usually marked 4.43319MHz or something similar. Strip and tin one end of the wire,

The second option is to switch off the colour crystal. This can be made switchable. After months of thinking about it (my soldering is lousy) this was the course I embarked on.

then solder this to one of the crystal tails. Run this wire up to the switch and cut it as short as is practical.

Strip and tin the end of this wire, then solder it to the vacant terminal on the switch.

Strip and tin one end of the remaining wire and solder it to the other tail of the colour crystal.

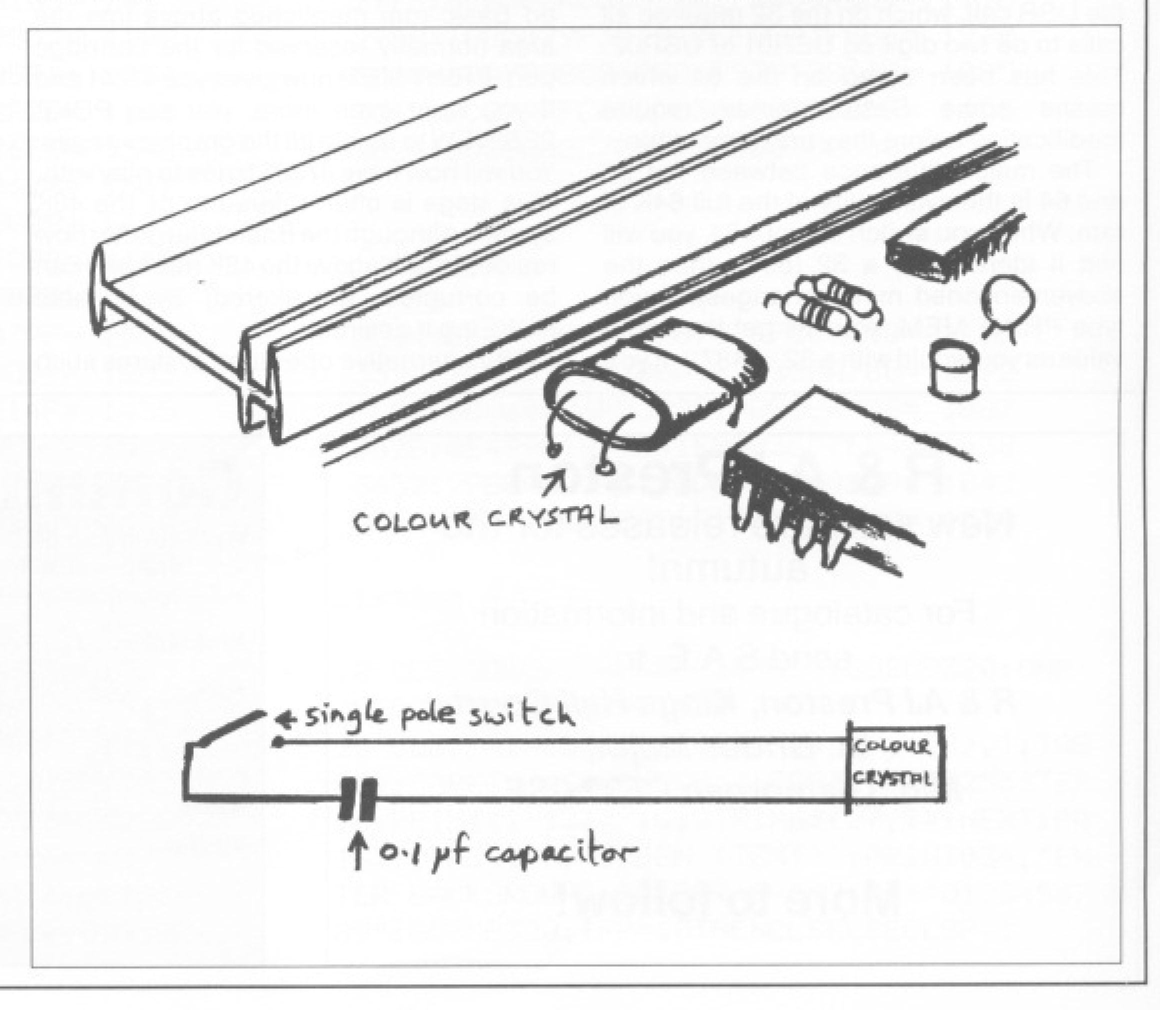
Run this wire to the free tail of the capacitor. Again cut the wire as short as practical. Strip and tin the end, then solder it to the free tail of the capacitor.

Some points to remember are: don't use an electrolytic capacitator. These have to be connected to the correct polarity or they go pop; don't use co-axial cable, which acts like a capacitor on its own so that the switch will be useless. On some machines the area around the crystal is so crowded, it might be easier to remove the main board from the machine and attach the wires to the underside at the point where the crystal is soldered into the board. Finally, don't forget to unplug everything but the soldering iron before you start. All that now remains is to plug in and switch on.

Even TV

The result should be a much smarter picture, both on text and Hi-Res screens and being switchable it will not affect the programs that require colour. In fact the display in PMODE 4 is so much improved it would even be worthwhile for those using colour televisions.

My thanks go to Ted Bacerelli who provided the information and to Les Gutteridge who was kind enough to respond to my letter on the subject. The funny thing is that Ted told me it was common knowledge, so why were so many of us still in the dark? Perhaps someone else has some secrets they could share.



Duplidisk update

***Graham Smith** puts Duplidisk on disc.*

AS all you faithful *Dragon User* readers will know, Quickbeam no longer exists. Computape now have the copyright on all the ex-Quickbeam titles. Orange Software have negotiated a licence with Computape which allows us to market the ex-Quickbeam range on disc.

One title which many people will already have purchased is *Duplidisk 2*, the tape to disc transfer utility. Many of you will also have found that it does not transfer itself to disk, which means that every time you want to use it, you have to load it from tape.

We have now produced the following two utilities to allow you to put your *Duplidisk 2* on disc. One deals with DragonDOS and the other with DeltaDOS.

Obviously, we will be more than happy to supply working copies on disc (£1.99 + 50p p&p) if you don't fancy typing these in. We will also be happy to supply the whole *Duplidisk 2* utility on disc for £9.45 plus 50p p&p, for those of you who haven't already bought a cassette version. Remember to specify DragonDOS or DeltaDOS.

On with the proggies. They are listed in Basic. You type these in and then RUN them to produce the machine code converter utility. You are given the START, END and EXEC locations, so you can save them to disc if you want to.

The procedure is them simply to position your *Duplidisk 2* cassette in your cassette recorder. EXEC your utility and press PLAY on the recorder. The utility will load the *Duplidisk 2* without allowing it to run. It then tells you the relevent locations for saving the *Duplidisk 2* to disc, and that is it.

You then have a disc-based version of *Duplidisk 2*, which you can RUN as required (or RUMN in the case of DeltaDOS). If you want any further details, send a stamped SAE to Orange Software, The Garth, Star Road, Nant-y-Derry, Avergavenny, Gwent NP7 9DP.

```
10 CLS:PCLEAR4:S=&HC00:E=&HD0F:X=&HC7E
20 PRINT"DUP2DEL-DUPLIDISK 2 TAPE TO"
30 PRINT"DELTA DOS CONVERTOR"
40 PRINT"START"S
50 PRINT"  END"E
60 PRINT" EXEC"X
70 FOR I=S TO E STEP16:CS=0:READ D$
80 FOR N=1 TO LEN(D$)/2
90 DD$="&H"+MID$(D$,(N*2)-1,2)
100 IF DD$="&H**" THEN 150
110 D=VAL(DD$):CS=CS+D:POKE(I+N-1),D
120 NEXT N:READ CS$
130 IF CS=VAL(CS$) THEN NEXT I ELSE PRINT"error"
140 END
150 READ CS$:IF CS<>VAL(CS$) THEN PRINT"error"
160 DATA 7FFF487FFFDE1A108E03E89F959F688E, 2190
170 DATA 015E8639A7808C01D125F910CE03A38E, 1747
180 DATA 0F00CE0400EC81EDC18C110025F78E12, 1621
190 DATA 00CE01D1EC81EDC18C134025F78E1400, 1880
200 DATA CE0600EC81EDC18C146025F78E1500CE, 1916
210 DATA 6E30EC81EDC18C1CE025F78E1D00CE76, 2124
220 DATA 00EC81EDC18C1FA025F78E2000CE2710, 1845
230 DATA EC81EDC18C21E025F71CEF7E27108E10, 2082
240 DATA 005FBDBAECBDB8B38E1000BDB7488E0F, 2017
250 DATA 006F808C300025F9860DBDB54ABDBDCF, 1889
260 DATA CE0D01AEC1271734409F7EBDB93E2618, 1548
270 DATA 967C810127F381FF260E354020E5BD80, 1817
280 DATA 188E0CE5BD90E539BD80188E0CD2BD90, 2064
290 DATA E520FE54415045204C4F4144494E4720, 1387
300 DATA 4552524F5200736176656D204455504C, 1275
310 DATA 49322C2648304330302C264832314530, 858
320 DATA 000F001200140015001D002000000**
330 DATA  135
```

```
10 CLS:PCLEAR1:S=16128:E=16374:X=16235
20 PRINT"DUP2DISK-DUPLIDISK 2 TAPE TO"
30 PRINT"DRAGON DOS CONVERTOR"
40 PRINT"START"S
50 PRINT"  END"E
60 PRINT" EXEC"X
70 FOR I=S TO E STEP16:CS=0:READ D$
80 FOR N=1 TO LEN(D$)/2
90 DD$="&H"+MID$(D$,(N*2)-1,2)
100 IF DD$="&H**" THEN 150
110 D=VAL(DD$):CS=CS+D:POKE(I+N-1),D
120 NEXT N:READ CS$
130 IF CS=VAL(CS$) THEN NEXT I ELSE PRINT"error"
140 END
150 READCS$:IFCS<>VAL(CS$) THENPRINT"error"
160 DATA 7FFF487FFFDE8E03E89F959F688E015E, 2243
170 DATA 8639A7808C01D125F910CE03A38E4000, 1716
180 DATA CE0400EC81EDC18C420025F78E4200CE, 1909
190 DATA 01D1EC81EDC18C436D25F78E4400CE0C, 2033
200 DATA 00EC81EDC18C456325F78E4600CE2737, 1899
210 DATA EC81EDC18C47E425F78E4800CE7530EC, 2339
220 DATA 81EDC18C51A025F77E27378E40005FBD, 1934
230 DATA BAECBDB8B38E4000BDB748860DBDB54A, 2215
240 DATA BDBDCFCE3FEAAEC1271734409F7EBDB9, 2292
250 DATA 3E2618967C810127F381FF260E354020, 1395
260 DATA E5BD80188E3FC8BD90E539BD80188E3F, 2140
270 DATA B5BD90E520FE54415045204C4F414449, 1720
280 DATA 4E47204552524F520073617665204455, 1191
290 DATA 504C49322C2648334630302C26483531, 906
300 DATA 41302C26483346303000400042004400, 682
310 DATA 46004800000**
320 DATA  142
```

Crossword

Please get your answers in to *Dragon User Crossword* Department by *Dragon User*, the end of the month on the front cover.

DRAGON QUIZ

by Terry and Derek Probyn

All this month's answers are names of Dragon software. When the crossword is complete, the column marked with an arrow will spell out a phrase.

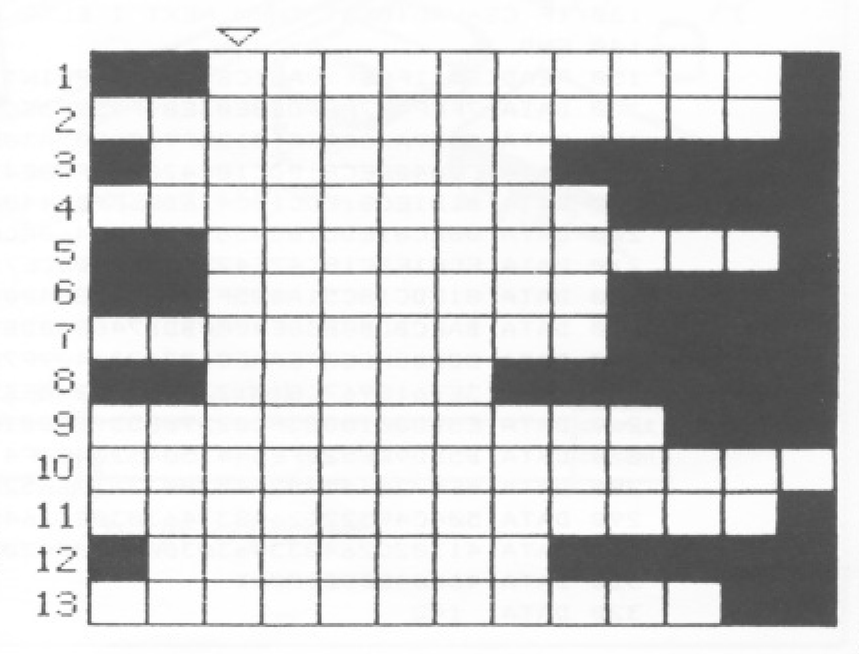
The tenth Dragon Crossword wonders what sort of hour in the morning this is, as the milkman rolls up yawning and dumps a bottle of snoring milk on the doormat. There is a message taped to the bottle. It is from Gordon Wright of Dunblane. 'This is the only text adventure I have ever solved' says he. "Do you have *Music Maker* in your Bottomless Box? Or Chuckie Egg?" I can't hear anything, Gordon ... likewise Brian Thomas of Chesterfield wants *Football* or *Airball*. What a load of games, Brian. We will see what we can do.

The phrase is TEXT ADVENTURE.

There will be a couple of free tapes from the Editor's Magic Bottomless Box for the first correct entries out of the hat each month. You can try telling us which tapes you'd like — you never know, we may have them.

And you don't have to cut up your *Dragon User* — entries on a photostat or a plain piece of paper will do, as long as we can read them.

1 Baby talk — where he dips his soldiers! (7,3)
2. Massive pothole which could be cool as calves! (8,4)
3 and 4 Formula One comes to Tangier? (7,5,4)
4. see 3.
5 and 8 Toy model bride in a spin, for big race destruction (10,5)
6. Don't despair, balloons are also one of these (7)
7. Keep calm! You cannot dip this oddly enough (4,5)
8. see 5.
9. For this game, put your money on the ham! (10)
10. If cod and pike own the company, there's dirty work afoot! (5,8)
11. Preface sight about future offensive craft. (5,7)
12. Sounds like eternal repose could lead to highest peak. (7)
13. Up to mischief again — so hangs insane in trouble! (11)



Write: ADVENTURE

***Pete Gerrard** cracks down on adventure utilities.*

Following on from last month's article, I read a review (in another magazine and for another computer, but the point remains the same) of an adventure writing utility. The point being made in the review was that with the aid of this utility it was possible to write superb adventures of a standard to match any commercial adventure currently on the market. Well, this being a family magazine I am prevented from using the word that I would like, but spherical objects comes fairly close. A great myth seems to have been built up about such utilities, and although reasonable games can be written with them, as many people have proved, what usually happens is that a flood of mediocre games is released instead. Indeed, I've lost count of the number of parodies of *The Hobbit* or *Colossal Cave* that I've seen, and never wish to see another one.

The only real point in favour of any utility that I've seen on any computer is that you don't have to produce your own routines like SAVE and LOAD, they're built in for you. GET ALL and DROP ALL are usually catered for, as are RAMSAVES and RAMLOADS. Graphics too are handled more often than not, although you've still got to draw the things in the first place, and if you're an artist like me then you get someone else to do them for you. The final great advantage is that the parser is already constructed, and if it's a sensible utility then it will be able to handle that which we covered in the very first article on writing adventures for your Dragon: sentences like OPEN THE RED CABINET AND LOOK INSIDE IT, although if it could cope with OPEN RED and LOOK IN it would probably do just as well.

Cut it out

A little hint here, while playing adventures rather than writing them. A friend of mine was playing a particular game, and wanted to cut a coupon out of a newspaper using some scissors. However, the adventure in question could only handle a VERB ... NOUN format, so CUT COUPON OUT WITH SCISSORS was clearly out of the question. In cases like this, said friend followed an interesting procedure. Write the sentence out that you want to type in, even though you know it won't be accepted by the adventure. Then try every possible two word combination until you find one that works. In this instance, I don't think anyone would have come up with the right input unless they had followed just such a procedure. Well, would you have thought of CUT OUT if I hadn't pointed you in the right direction? It's a good rule to follow when playing adventures that are restricted to VERB ... NOUN. But I digress, as usual, so back to adventure writing utilities and their claims to fame.

One of the chief of these seems to be that you don't have to learn to program to use them. Nonsense! I've been converting a game recently, off and on, that was written on just such a utility and trying to get it to run in good old Basic. Well, instructions like:

if verb="help" then lf message 306 wait

I can just about cope with. If the verb typed in is equal to "help" then print a line feed, print message number 306, then wait for another input from the player. I have a standard way of printing out messages, so I could easily convert the above into something like:

if vb=43 then print:mess=306:gosub 5990:goto 10



Here we're just looking for a verb number rather than a specific verb, we use the Basic keyword PRINT instead of 'lf', and use a subroutine to print out the message rather than the utility itself. Finally we go to line 10 instead of using the word 'wait' to wait for another input. That sort of thing is fairly straightforward, and variations on it were found throughout the game in question. Other commands could reasonably well be impersonated as well:

if verb="i" and notzer? cntobj with then lf message 9916 list with message 215 wait

As with printing messages I have standard routines for doing an inventory, and all that the above is saying is that if the player is carrying at least one object then print message number 9916 and list all the objects being carried. Later on there would be another line for handling an inventory request if nothing was being carried. But if you just look at a line like the one above, is it any wonder that I require a great pinch of salt when I read claims like "you don't need to know how to program"? Good grief, it's almost as complicated as learning machine code! Well, almost ...

But to give one final example from the game in question, what on earth would you expect to do with a command like this:

if firstob with then repeat lf message 1020 itis firstob with objsht firstob with drop firstob with message 1021 pause 25 until zero? firstob with

That sort of thing is just ridiculous, but it really is taken from an adventure game written with a utility and I have typed it in exactly as it is shown on the listing which is currently to my left as I write. No non-programmer is going to get to grips with commands of that complexity, at least not straightaway, so the whole point of the thing is: do not necessarily be seduced by advertisements for utilities that tell you that you don't need to learn to program. Read that above command line again and ponder on the wisdom of such a statement.

A good routine

But this is not to knock utilities altogether. If you don't program, and want to have a lot of the routines essential for an adventure game already written for you, then they do have a lot going for them. Particularly if you are approaching a computer for the first time, and if you think people are no longer approaching Dragons then read this month's *Adventure Trail* to find a 67 year old who is doing just that.

In the land of COD (Computers Other than Dragon), especially the new-ish range of 16 bit monsters, I doubt whether any one person could ever get sufficiently to grips with the machine to be able to write a brilliant adventure on it. In such a case, if a specific adventure writing utility exists then you might as well use it and save months and months of time learning how the machine works. Take a few weeks to learn how the utility works instead, and spend the rest of your precious time developing and writing the game. That's what I'm doing! The problem then, and this applies to Basic adventures on the Dragon as much as it does to utility ones on anything else, is to make the game different and to stand out from the crowd. A good utility allows its use to be hidden from the player, and a good adventure writer on the Dragon disguises the fact that he's written 90 per cent of a game in Basic. The problem remains the same.

Copy cats

Which brings us nicely to the remainder of this article, and a little chat about disguising adventures.

I should imagine that anyone setting out to write an adventure for the very first time has played at least one game from the genre, and thus has an idea of the sort of problems that are being posed in them. I remember my first game, a very humble affair, written after I had played *Colossal Cave* and a couple of Scott Adams games. Well someone has to! Anyway, in that humble adventure all I really did was to replace traditional problems with variations on them. Giving the eggs to the troll became giving a bottle of whisky to an old-timer. Waving a rod to produce a bridge was replaced by ... waving a rod to produce a bridge. Original that, I was proud of that one. Other problems were mostly variations on a theme, but by the fourth or fifth adventure I had progressed far enough along the scale to produce original and different problems of my own, with only the occasional borrowing from another game, and that was usually in the form of inverting the problem.

That's probably as good a place to start as any, inverting problems, and whether you're using a utility or not try and stay away from messages that are common to every other adventure in the world. Something like "You can't do that" can easily be replaced by something else, as can "I don't understand that sentence" or "You're using a word I don't know", both of which seem to crop up in just about every game under the sun.



Oops

So do status lines at the top of the screen, although there's no great harm in having them, as they do pass on a lot of useful information to the player. The phrase "What now" or "What next" could probably be got rid of, replacing it with a simple prompt such as "instead. And something which all adventures should have, and all too few do, is an "OOPS" command, that allows you to take back your last move in the event of something disastrous happening. How to implement such a thing? Well, if you've got a RAMSAVE routine then you're just about there. Simply perform a RAMSAVE after each input but before acting on it, and then if the player does have a disaster and the next input is "OOPS" then you just call back the last RAMSAVEd position and the player can try again. Easy, isn't it? Bye for now.



A thousand apologies to devoted readers of this column for having in last month's pages what we in the trade refer to as a technical disaster. One could conjure up a myriad excuses but one won't. Avid collectors of the Adventure Trail will now have two solutions to the *Pyramids of Doom* adventure, and for that one can only offer a multi-lingual muchos apologieso. A straightforward solution and a Professor Deadrock solution have now been printed, and I promise never to refer to that particular adventure again. I hang my head in shame ...

But I quickly pick it up again and peer into this month's mailbag.

First off is a letter from our old friend Nick Hodge. Like many people he has completed a formidable list of adventures for which he is offering help should any require it, and these adventures include *Trekboer, Aquanaut 471, Vortex Factor, Juxtaposition, Syzygy, The Ring of Darkness*, and *Pettigrew's Diary*. Anyone wanting to get in touch should write to Nick at Caramic, Huntworth, Bridgewater, Somerset TA7 0AJ.

An interesting point arises in Nick's letter, and this is that he is finding it increasingly difficult to get hold of some of the older but still fascinating Dragon adventures. He would be willing to swap them for adventures that he's completed, or buy them if necessary, so perhaps we could start some sort of global Dragon adventure software swap shop if anyone's prepared to set the ball rolling. Adventures that young Hodge is keen to lay his hands on are *Madness and the Minotaur, Sea Quest, Shenanigans, Sanctum, Calixto Island, Total Eclipse, Caverns of Doom, White Cliffs of Dover, The Ket Trilogy, Death Mines of Sirus*, and (it says here) last but by no means least *El Diablero*.

If you've got Dragon adventures for sale, or are looking for a particular title, then drop us a line and I'll do my best to include you in the *Adventure Trail* as soon as time and deadlines allow. Just imagine, Dragon adventures whizzing up and down all over the country! But no piracy, chaps and chapesses, let's stick to original copies only. Keep the backups for your own personal use, as always, and deal exclusively with originals. I'm sure there's something wrong in that particular line of logic, but we'll ignore it and just repeat that we do not want to see any piracy going on.

Any road up, as we Northerners are wont to say, Nick has more than a few hints concerning a natty little number called *Return of the Ring*, so without further ado we read on and find that he's nearly finished the game (he thinks), and this is how he's managed to get as far as he has ...

1) Find the stone Xandra in the Room of Doors and give her the staff.

2) When she reappears on the planet take the flask from her, go to the genie, and then type in the instruction Drop Flask.

3) Go to Hamles chest and open it with the genie's key. Get the treasure sack.

4) Take the sack to King Cebar then take it to the Lost Krell. Leave it there, and when you return you'll find a unit.

5) Get the eviction spell and cast it in the Trog's cave before taking the amulet. Now then, said Peter, interrupting sternly. A certain sprightly 67 year old called Jim Finley, from Romford (curse those bitter drinking adverts) tells me that the spell is called the

Banishment spell. We shall return to Jim ere too much water has passed under the bridge. Or something like that.

6) Get fake book from Oracle and go to Nightfall Mutant's cave and drop that very same fake book. Get the book of skulls and take it back to the Oracle.

7) Take stone, black orb and cyclop's eye to the relevant Room of Many Quests (sounds like something out of the *Krypton Factor* — go to the room of many quests and there you will find ... seventy three pieces of see-through plastic which have to be assembled into a working model of a high tech. number crunching telephone). Where was I? Oh yes, take all that to the Room of Many Quests and you'll get a little something for your troubles.

8) Get the sack of magic grain from the grain store and take it to King Cebar, who appears to be getting an awful lot of goodies in this game.

9) When you have the six units (hah!) take them to the time chamber and bind them. This will get you a Time Ring.

Small and blue

All well and good, but that's as far as Nick has got. What does he do with the time ring, how does he pick it up, why can Xandra pick it up when he can't? These and many other questions are also asked by one Keith Porteous of Hackenthorpe in Sheffield, who writes on the smallest sized stationery that I've ever seen. About four inches by three (I refuse to go metric) and coloured a delicate shade of blue. What can it all mean? With an address like Hackenthorpe he ought to be able to take the code of the game to pieces and find out that way, but perhaps not. Self-same problem, apparently only the ring bearer can take the ring, which brings us back to the aforementioned Jim Finley of Romford.

No, he's not a ring bearer, and is not essential to the completion of the game. He probably wouldn't fit in the packaging anyway. However, he is writing about the same game as our friend Nick, namely *Return of the Ring*, so let's take a look at what he has to say.





Nice start — "This is the first time you've heard from me as I'm a fairly new boy to the Dragon. (Boy? Who's he trying to kid? He's pressing on 67 years old)." At last, a reader who doesn't claim to be 5 years old and to have solved his first adventure before he was born. Apparently friend Jim is "hopelessly lazy" and has "waited until now to write in the hope that someone else has done all the work anyway!" I too live in a dream world at times, especially on Monday mornings, when you turn the computer on and just stare at it in the hope that it will do something. You sit there and think "come on, you do something for a change, it's always me that does all the typing, you do it for once". It rarely does. Anyway, back to Jim's letter and his own discoveries on *Return of the Ring*.

On your own

"Leaving the town and heading north-north-west will bring you to the oracle. From there due north will bring you to the time chamber, where you need to deposit six units. From there you're on your own because I've only found five so far (thanks, friend — PG). If you leave the Shedir ring and take the other four to the amplifier you can trade three of them for maximum strength, charisma and intelligence (why don't these things work in real life?!). The fourth you can carry into the moon forest where, as you will, 'ring use, mutants lose', and you will avoid falling down traps."

"Genies like living in bottles and this poor genie is in a flap because he's broken his so give him another, or something like it: a flask will do. You haven't got a flask? Ask the Princess. Deposit the flask in the cave and go out. Then come back to find a happy re-housed genie and a key on the floor. This technique is helpful elsewhere: HINT!"

"To get rid of the Trog, banish him; for that you will need a spell of banishment (or eviction, if you're Nick Hodge — PG)". Slight confusion then follows in Jim's letter, as he can't remember where the spell of banishment is, so we'll skip over that bit and carry on later. "I've a note that the Healer is at the Temple of Regeneration. However, all that you can gain there is the restoration of lost lives, so if you haven't lose any, nothing will happen." There we reach the end of the help on *Return of the Ring*, but Jim does go on to tell me that some scurrilous rag which shall be unnamed has published a cheat for the game which gives you "unlimited pretty well everything. Why anyone should wish to take all the interest and sense of achievement out of the game I can't imagine" says Jim, and I totally agree. Hints yes, solutions



even, but if adventures go the way of arcade games and pages and pages are dedicated to ways of getting unlimited lives then we're in a sorry state indeed. Fine for arcade games, I defend The Expert to the hilt, but not for adventures.

A quickie to finish with. A chap called Rob Elwes, from Middlesex, has a problem with the *Vortex Factor*, but fails to actually state what the problem is. He has all the cartridges, has got past the "Cairo bit and the post nuclear accident bit", and then has ground to a halt. Well, without knowing what the problem is not even this mighty column can give you a solution. We need more details!! Bye for now.

Adventure Contact

To help puzzled adventurers further, we are instituting an Adventure Helpline — simply fill in the coupon below, stating the name of the adventure, your problem and your name and address, and send it to Dragon User Adventure Helpine, 49 Alexandra Road, Hounslow, Middx TW3 4HP. As soon as enough entries have arrived, we will start printing them in the magazine.

Don't worry — you'll still have Adventure Trial to write to as well!

Adventure ....

Problem ....

....

Name ....

Address ....

....

A little number

***Gordon Lee** finds that slicing up pi is not simply a piece of cake.*

INVESTIGATIONS into the transcendental number pi have had a long and chequered history a period of some five thousand years since it was first realised that there was more to this enigmatic number than meets the eye. Its value is simply the ratio of a circle's circumference to its diameter. However, nothing is 'simple' when it comes to evaluating this number. The term 'transcendental' gives a clue to its intriguing nature, that is, it has a decimal value which extends to infinity without recurring or repeating. Its true value has now been computed to over one million decimal places, although there can be no practical use for such a task!

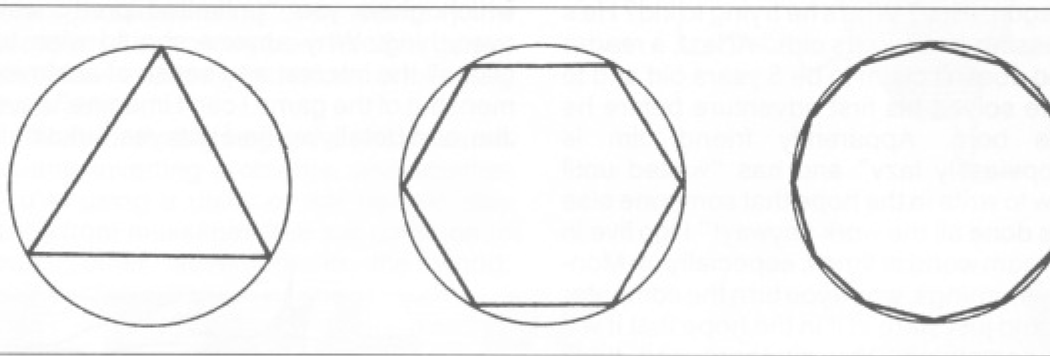
For most purposes a value to six or seven decimal places is more than adequate, but if you require a greater degree of accuracy, here is pi to 35 decimal places:

3.14159 26535 89793 23846 26433 83279 50288

The problem is, of course, remembering such a sequence, and over the years a number of mnemonics have been developed sentences in which the number of letters in each word corresponds to the digits in pi. 'How I hope I write logically so Dragon users can solve problems enjoyably (ontheir) computers' would be one such mnemonic. OK, so I cheated a bit by running two words together near the end. A much better attempt was written by Adam C. Orr of Chicago in 1906:

Now I, even I, would celebrate
In rhymes unapt the great
Immortal Syracusan rivaled nevermore,
Who in his wondrous lore,
Passed on before,
Left men his guidance,
How to circles mensurate.

Note the American spelling of 'rivaled' to ensure a '7' for this digit. The 'Immortal



PERIMETERS of INSCRIBED and CIRCUMSCRIBED POLYGONS

s	Length of side (Internal)	Length of side (External)	Perimeter (Internal)	Perimeter (External)
3	0.8660254038	1.154700538	2.598076211	3.464101614
6	0.5	0.5358983846	3	3.215390308
12	0.258819045	0.2633049951	3.10582854	3.159659942
24	0.1305261922	0.1310869256	3.132628613	3.146086213
48	0.06540312921	0.0654732208	3.139350202	3.142714598
96	0.03271908282	0.03272784426	3.14103195	3.141873049
192	0.01636173162	0.01636282681	3.141452472	3.141662747
384	0.008181139603	0.008181276498	3.141557607	3.141610175
768	0.004090604024	0.004090621137	3.141583891	3.141597033
1536	0.002045306292	0.00204530843	3.141590462	3.141593748
3072	0.00102265368	0.001022653948	3.141592105	3.141592926
6144	0.000511326907	0.0005113269405	3.141592517	3.141592721

Prize

The wheel is come full circle. Can it be that time of year again? Not quite - but looking ahead, Dragonfire Services are offering ten free entry tickets to the Colour Computer Convention in Weston-super-Mare in November - see *Newsdesk* for further details. Those winners who know that they can't make the show can opt to take their chances with the Editor's Magic Bottomless Box. A piece of pi.

Rules

When you have calculated the Great Unknown and arrived back at the place you started from, put your conclusions, your listings and the famous tiebreaker into an envelope marked SEPTEMBER COMPETITION and send it to us.

As for the tiebreaker, perhaps you should all devise a mnemonic for pi in no less than seven lines of rhyme, including at least one ancient Greek ... but no, I will spare you. Mr. Orr died some time ago, so poetry can't have been good for him. Instead, think of yet another way of misspelling a common word (such as rivaled or ontheir) to help you remember something. Now, juno which puzzler we're judging this month?

June winners

Lots of entries to this competition, with a high proportion of right answers. Few people broke the coded message, though - see across the way for that. The 1819 crew are:

C. Hitchinson of Middlesbrough, D J Gray of Middlesbrough, J D Hartley of Cleckheaton, D R Sharples of Merseyside, T H Denton of South Norwood, Ronald D Walters of Walsall, R M Cashmore of Market Harborough, Austan Henderson of Bromsgrove, John S Blatch of Weybridge and F J Taylor of Middlesbrough. All these will receive copies of *Five Games Tape* from **Preston Software** who, incidentally, have some good new games on the market.

We had some excellent, practical suggestions for promoting the Dragon, including car stickers, local contact points and bounty for readers introducing new subscribers. The most ambitious one was Austan Henderson's "Get someone to hack into the national newspaper computers and create an automatic full page ad. without anyone noticing."!

Solution

See opposite.

Syracusian' referred to in this rhyme was the Greek mathematician and geometer, Archimedes, who was one of the earliest to attempt to calculate the value of pi. The method that he used involved the calculation of the perimeter of regular polygons both inscribed and circumscribed about a circle of diameter 1 unit. In **figure one** the length of each side of the inscribed triangle can be easily calculated using an extension of Pythagorus' theorem. As the circle has a diameter of 1 unit, its circumference will have the value pi, and consequently the perimeter of the triangle will give a (very) rough approximation to this value. By doubling the sides of the polygon, each successive calculation in the series will give a closer and closer approximation to the required value. If a similar series of calculations is performed but using *circumscribed* polygons, then the true value of pi will lie somewhere between each pair of values. The problem of using such a method lies in the calculation of a series of square roots, each successive doubling of the number of sides in the polygon involves a more intricate and lengthy root to be evaluated. The table (**figure two**) shows the results of such an operation on polygons with dies in the doubling series of 3 to 6144. Archimedes calculated as far as a 96-sided polygon, but in the talbe the values have been extended a little further.

From his calculations, Archimedes was the first to give the approximate value of 22/7 for pi, a figure that is still used today when only a rough computation is required. Three thousand years before Archimedes the Babylonians were using three-and-one-eighth in their calculations, while in Egypt c. 1800 BC, the area of a circle was calculated as being the square of 8/9ths of its diameter. Ch'ong Hong (79-139) believed pi to be equal to the square root of 10, while Aryabhata (476-550) gives it the remarkably accurate 3.1416. At about the same time, the Chinese engineer Tsu Ch'ung Chi devised the amazing fraction 355/113. THis produces an accuracy of pi to six decimal places, accurate enough to compute the circumference of the earth, given its radius, to within eleven feet!

More recently, the Indiana State Legislature considered a bill in 1897 to regard pi as having a value of 3.2 exactly. The bill was defeated. On a more practical note, mathematicians have frequently devised rational approximations for pi, but few have been able to exceed the Tsu Ch'ung Chi fraction for accuracy, and this value is probably the most useful for everyday calculations. In **figure three**, the table shows some of these approximations, the accuracy of each has been tested by using each value to compute the circumference of the earth. The difference from the true value is shown in the right-hand column. The final value in the table remains blank as this forms the basis of this month's competition. In 1914, the Indian mathematician Srinivasa Ramunujan devised a rational approximation which gives pi to an even greater accuracy. Using the test already described, this approximation will calculate the circumference of the earth to within one inch (assume that the earth has a radius of 3960 miles, and therefore a circumference of 2*pi*3960). The approximation that Ramanujan found is given as the square root of the square root of the value indicated in the brackets (?). This unknown, which is what you have to find, is a fraction with a whole number for both the numerator and the denominator. What are the simplest numbers which will fit?

Approximation		Decimal value	Accuracy
Biblical		3	-1121 miles
John Lambert	$(7/4)^2$	3.0625	-626 miles
Indiana State Legislature		3.2	+463 miles
Ch'ong Hong	$\sqrt{10}$	3.16227766	+164 miles
Egyptian	area=$(8d/9)^2$	3.16049383	+150 miles
Babylonian	$3\frac{1}{8}$	3.125	-131 miles
John Lambert	$(62/35)^2$	3.13795918	-29 miles
Archimedes	22/7	3.14285714	+10 miles
Aryabhata		3.1416	+307 feet
Tsu Ch'ung Chi	355/113	3.14159292	+11 feet
Srinivasa Ramanujan	$\sqrt{\sqrt{(?)}}$		<1 inch

The Answer

This is Gordon Lee's own solution to the June competition see page 26 for results

Puzzle one

ANSWER: the smallest number which 'hailstones' to a maximum in excess of one million is 1819. This starting value reaches a maximum of 1276936, the whole sequence taking 61 ateps to reduce to 1.

Listing one runs the test on nall numbers from 1 upwards, each time computing the path length (P) and the maximum value reached (M). As each run is completed the results are printed out. This is continued until the maximum printed at line 180 exceeds one million.

In the program, note that it is only necessary to test for a new maximum after an odd number has required the value to be multiplied (lines 150 and 160).

Puzzle two

ANSWER: the quotation was 'The only competition worthy a wise man is with himself' (Mrs Anna Jameson, 1794 - 1860). The code used was a substitution code using a 'key' to denote the substitution required for each letter. Clearly, the phrase given was unlikely to have been in a straight substitution code as, in this case, the three-letter sequence EEF would indicate a word of three letters, the first two being the same. As this is unlikely, the use of a key is indicated.

This key is in fact the words 'Dragon User', used repeatedly throughout the message. Each letter in the quotation was advanced by the number of letters indicated by its corresponding 'key' letter D = 4 as D is the fourth letter of the alphabet.

THE ONLY COMPETITION WORTHY A WISE MAN IS WITH HIMSELF
DRA GONU SERDRAGONUS ERDRAG O NUSE RDR AG ONUS ERDRAGO
XZF VCZT VTETWUPIWJG BGVLIF P KDLJ EEF JZ LWOA MAQKFSU

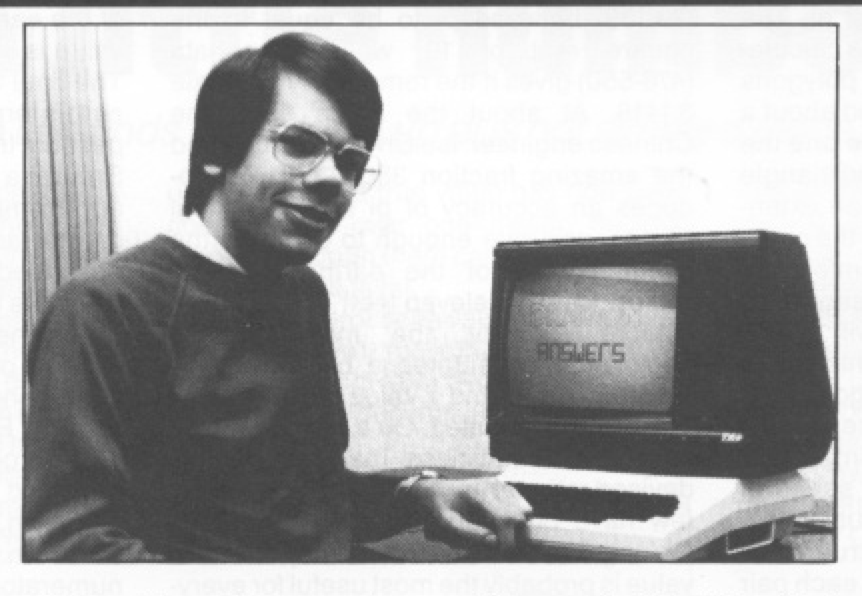
So, T plus D (4) will give X, and so on. To decipher it (provided that you know the method to use!), the process is reversed. This is done using **listing two**, which uses the ASCII values of the letters as a basis for the substitution. Line 150 ensures that all 'non-letters' (spaces, punctuation marks, etc.) are left unchanged.

```
Listing 1
>
  100 START=1
  110 N=START
  120 P=0:M=N
  130 IF N=1 THEN 180
  140 IF N/2=INT(N/2) THEN N=N/2:GOTO 170
  150 N=N*3+1
  160 IF N>M THEN M=N
  170 P=P+1:GOTO 130
  180 PRINT START;"  ";P;"  "M
  190 IF M>=1000000 THEN STOP
  200 START=START+1:GOTO 110
```

```
Listing 2
>
  100 D$="DRAGONUSER"
  110 M$="XZF VCZT VTETWUPIWJG BGVLIF P
  KDLJ EEF JZ LWOA MAQKFSU."
  120 Z$="":D=1
  130 FOR F=1 TO LEN(M$)
  140 A=ASC(MID$(M$,F,1))
  150 IF A<65 OR A>90 THEN 180
  160 A=A-(ASC(MID$(D$,D,1))-64):D=D+1
   :IF D>10 THEN D=1
  170 IF A<65 THEN A=A+26
  180 Z$=Z$+CHR$(A)
  190 NEXT
  200 PRINT M$:PRINT Z$
```

Dragon Answers

If you've got a technical question write to Brian Cadge. Please do not send a SAE as Brian cannot guarantee to answer individual inquiries.



Double your notes

I have a Dragon 64 and Microdeal's *Composer* program. Do you know if there is any way of using the 64's extra memory to be able to produce longer musical pieces, as I am finding the 720 note maximum more and more restricting?

Joan Blackburn
66 Ince Green Lane
Ince-in-Makerfield
Wigan
Lancs WN2 2AR

THE first few lines of the Composer Basic program set where the music program and data are stored and the maximum number of notes allowed. By default, the machine code starts at address $6000, and the compiled music immediately after this at $6600 which gives space for 720 notes (the end of memory being set to $7FFF).

Operating in 64K mode allows an extra 16K of music, but remember that the mujsic is stored as DATA statements at the end of the program so it will be necessary to move the machine code and compiled music up in memory (this is easy as the code is relocatable, but it must lie on a 256 byte boundary). Adjusting the CLEAR statement to reflect this will allow the extra DATA statements to fit.

By experimenting with different settings, you should find that you can get about twice as many notes in memory with the extra 16K.

Sort it out

I have written a program in Basic, part of which has to sort out about 400 short strings into alphabetical order, having loaded them from tape. I am using the 'bubble sort' technique for simplicity. The program works fine for a while, then for no apparent reason it 'hangs up' and the BREAK key does not respond. After pressing RESET and examining the partly sorted list, some of it has been turned into meaningless strings of letters!

Can you explain what is happening, is this task just too much for my Dragon 32?

John Smallwood
51 Kings Drive
Fulwood
Preston
Lancs PR2 3HQ

WHEN working with a large number of individual strings (in your case 400) the Dragon has to occasionally do a 'garbage collection' of string space. This occurs becasue swapping the values of two strings does not move the strings of characters themselves, just the pointers to them in string space. Thus with strings of different lengths small segments of unused space build up the string space becomes fragmented.

The more string swapping there is, the more fragmented the string memory space becomes. As the bubble sort uses an awful lot of string swapping, the memory soon becomes too fragmented to use.

The 'pause' in your program is the Dragon re-organising its string space, moving strings around so that these small fragments disappear freeing more space. Pressing RESET in the middle of this will leave some string variables pointing at the wrong addresses and hence meaningless data.

The answer is either to be patient or to use a better sorting algorithm than the bubble sort.

Inside-out

ABOUT two months ago I wrote a simple program to calculate the missing side of a triangle when given the hypotenuse and the other side. However, when I used the program a week later, sometimes it gave an FC Error in line 60 which reads:

60 SI=SQR(HYP*HYP-AB*AB)

Although sometimes the program works OK. I was wondering if you could tell me why this happened as I am very curious.

Michael McCullagh
126 Cregagh Road
Belfast BT6 0LA

THE SQR (square root) function will give an FC error if its parameter is a negative number as it is not possible to find the square root of any non-positive number. In your program, this only happens if the value of AB is greater than that of HYP, which it should never be as the hypoteneuse of a triangle is, by definition, the longest side.

Add a line to check that the values entered are valid before calculating the third side, such as:

55 IF AB>HYP THEN PRINT "HYP MUST BE LONGEST SIDE":GOTO 10

Dumb ascii

I would like to use my Dragon 64 computer as a dumb ascii terminal, communicating through the RS-232 serial port of the machine. The manual only mentions the use of DLOAD and DLOADM for downloading Basic and machine code programs between two Dragons. I would be very grateful if you could kindly give me the information needed to write a terminal program.

Y. Mandel
P O Box 651258
Benmore 2010
South Africa

THE information regarding the serial port has been printed here before, but as this is only one of a number of letters requesting this information from the elusive 'Additional Information' booklet, I'll give it again, together with machine code versions for assembler programmers...

Set Baud Rate:

```
POKE &HFF07,(PEEK(&HFF07)AND&HF0) OR B
```

```
SETB LDA  $FF07
     ANDA #$F0
     ORA  #BAUD
     STA  $FF07
     RTS
```

Where B is from 1 to 15 representing the rates; 50, 75, 110, 135, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600.

Send a character:

```
10 IF PEEK(&HFF05)AND16=0 THEN 10
20 POKE &HFF04,CH
```

```
SEND LDA  $FF05
     ANDA #16
     BEQ  SEND
     LDA  #CH
     STA  $FF04
     RTS
```

Wait for a character:

```
10 IF PEEK(&HFF05)AND8=0 THEN 10
20 CH=PEEK(&HFF04)
```

```
WAIT LDA  $FF05
     ANDA #8
     BEQ  WAIT
     LDA  $FF04
     RTS
```